



TESIS - SS142501

**METODE ENSEMBEL ROCK DAN SWFM
UNTUK PENGELOMPOKAN DATA CAMPURAN
NUMERIK DAN KATEGORIK
PADA KASUS AKSESI JERUK**

**ALVIONITA
NRP. 1315201008**

**DOSEN PEMBIMBING
Dr. Sutikno, S.Si., M.Si.
Dr. Drs. Agus Suharsono, M.S.**

**PROGRAM MAGISTER
JURUSAN STATISTIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2017**



TESIS - SS142501

**ENSEMBLE ROCK AND SWFM METHOD
FOR CLUSTERING MIXED
NUMERICAL AND CATEGORICAL DATASET
IN CITRUS ACCESSIONS CASE**

**ALVIONITA
NRP. 1315201008**

**SUPERVISORS
Dr. Sutikno, S.Si., M.Si.
Dr. Drs. Agus Suharsono, M.S.**

**PROGRAM OF MAGISTER
DEPARTMENT OF STATISTICS
FACULTY OF MATHEMATICS AND NATURAL SCIENCES
INSTITUTE OF TECHNOLOGY SEPULUH NOPEMBER
SURABAY
2017**


METODE ENSEMBEL ROCK DAN SWFM UNTUK PENGELOMPOKAN DATA CAMPURAN NUMERIK DAN KATEGORIK PADA KASUS AKSESI JERUK

Tesis disusun untuk memenuhi salah satu syarat memperoleh gelar Magister Sains (M.Si)
di
Institut Teknologi Sepuluh Nopember
Oleh:


ALVIONITA
NRP. 1315 201 008

Tanggal Ujian : 11 Januari 2017
Periode Wisuda : Maret 2017


Disetujui oleh:


1. Dr. Sutikno, S.Si., M.Si.
NIP. 19710313 199702 1 001

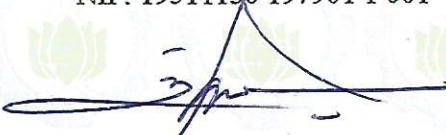
(Pembimbing I)


2. Dr. Drs. Agus Suharsono, M.S.
NIP. 19580823 198403 1 003


(Pembimbing II)


3. Dr. Drs. I Nyoman Latra, M.S.
NIP. 19511130 197901 1 001

(Penguji)


4. Dr. Dra. Ismaini Zain, M.Si.
NIP. 19600525 198803 2 001

(Penguji)


an. Direktur Program Pascasarjana
Asisten Direktur
Prof. Dr. Ir. U. Widjaja, M.Eng.
NIP. 19611021 198603 1 001

Direktur Program Pasca Sarjana,

Prof. Ir. Djauhar Manfaat, M.Sc., Ph.D.
NIP.19601202 198701 1 001

Metode Ensemble ROCK dan SWFM untuk Pengelompokan Data Campuran Numerik dan Kategorik pada Kasus Aksesori Jeruk

Nama Mahasiswa : Alvionita
NRP : 1315201008
Pembimbing 1 : Dr. Sutikno, S.Si., M.Si.
Pembimbing 2 : Dr. Drs. Agus Suharsono, M.S.

ABSTRAK

Salah satu permasalahan yang sering ditemui dalam analisis kelompok adalah data yang berskala campuran numerik dan kategorik. Metode untuk mengelompokkan data campuran diantaranya metode ensemble dan metode *Similarity Weight and Filter Method* (SWFM). Tahap pengelompokan metode ensemble dilakukan dengan algoritma pengelompokan data kategorik, salah satunya adalah metode ROCK (*RObust Clustering using linKs*). Terdapat banyak penelitian dan pengembangan mengenai kedua metode tersebut, namun penelitian mengenai perbandingan kinerja dari kedua metode masih terbatas. Oleh karena itu, penelitian ini melakukan perbandingan kinerja antara metode ensemble ROCK dan ensemble SWFM. Kedua metode digunakan pada studi kasus pengelompokan aksesori jeruk hasil fusi protoplasma yang merupakan data campuran numerik dan kategorik. Metode pengelompokan terbaik ditentukan dengan kriteria rasio antara simpangan baku di dalam kelompok (S_w) dan simpangan baku antar kelompok (S_B) terkecil. Berdasarkan 25 objek pengamatan pada studi kasus, metode ensemble ROCK dengan nilai θ sebesar 0,27 menghasilkan tiga kelompok dengan nilai rasio sebesar 0,1358, sedangkan metode ensemble SWFM menghasilkan dua kelompok dengan nilai rasio sebesar 0,3059. Hasil tersebut menunjukkan bahwa metode ensemble ROCK memberikan kinerja pengelompokan lebih baik daripada metode ensemble SWFM. Karakteristik hasil ensemble ROCK yang diperoleh adalah (a) kelompok 1 beranggotakan 10 aksesori dengan buah berukuran kecil tetapi berat, berkulit tebal, berwarna dominan kuning kehijauan, permukaan dominan halus, tekstur *pulp* beragam, dan berkadar air sedang, (b) kelompok 2 beranggotakan 7 aksesori dengan buah berukuran sedang tetapi berat, berkulit tipis, warna kulit beragam, permukaan beragam, tekstur *pulp* beragam, dan berkadar air tinggi, dan (c) kelompok 3 beranggotakan 3 aksesori dengan buah berukuran besar tetapi ringan, berkulit tebal, berwarna kuning kehijauan, permukaan halus, *pulp* lembut, dan berkadar air sedang.

Kata Kunci : Data Campuran, Data Numerik, Data Kategorik, Pengelompokan Ensemble, *Similarity Weight, Filter Method*, ROCK, Aksesori Jeruk.

“Halaman ini sengaja dikosongkan”

Ensemble ROCK and SWFM Method for Clustering Mixed Numerical and Categorical Dataset in Citrus Accessions Case

Name : Alvionita
Student Id. Number : 1315201008
Supervisor : Dr. Sutikno, S.Si., M.Si.
Co-supervisor : Dr. Drs. Agus Suharsono, M.S.

ABSTRACT

One of the problems often encountered in clustering analysis (cluster) is a mixed numerical and categorical dataset. The method is used to analyze the mixed dataset including ensemble method and Similarity Weight and Filter Method (SWFM) method. In the ensemble method, the clustering is done with categorical data clustering algorithm, one of them is a ROCK method (Robust Clustering using links). There is a lot of research and development concerning both methods, but research about performance comparative between methods is still limited. Therefore, this study do a performance comparison between the ensemble ROCK method and ensemble SWFM method. Both of these method are used for the case study about clustering of citrus accessions which have a mixed numerical and categorical dataset. Best clustering method is determined by the smallest ratio of standard deviation in groups (S_W) and standard deviation between groups (S_B). Clustering result for 25 observation obtained by using the ensemble ROCK method with values of θ is 0,27 produces three groups of data with ratio value of 0,1358, while the ensemble SWFM method produces two groups of data with ratio value of 0,3059. These results suggest that ROCK ensemble method provides better performance than the ensemble SWFM method. Characteristics of ensemble ROCK's results are (a) group 1 consisted of 10 accessions with a small but heavy fruit, thick-skinned, the color dominant greenish yellow, predominantly smooth surface, diverse pulp, and medium water content, (b) group 2 consisted of 7 accession with medium size but heavy fruit, thin-skinned, variety color, diverse surface, diverse pulp, and high water content, and (c) group 3 consists of 3 accession with large but lightweight fruit, thick-skinned, yellow-green color, smooth surface, soft pulp, and medium water content.

Keywords: Mixed Data, Numeric Data, Categorical Data, Clustering Ensemble, Similarity Weight, Filter Method, ROCK, Citrus Accession.

“Halaman ini sengaja dikosongkan”

KATA PENGANTAR

Namo Buddhaya,

Puji dan syukur penulis panjatkan kehadirat Tuhan Yang Maha Esa, karena dengan rahmat-Nya, penulis dapat menyelesaikan draft tesis dengan judul “**Metode Ensembel ROCK dan SWFM untuk Pengelompokan Data Campuran Numerik dan Kategorik pada Kasus Akses Jeruk**”. Tesis ini merupakan salah satu syarat kelulusan dalam pelaksanaan tugas akhir program pascasarjana strata-2 (S2) di Program Magister Jurusan Statistika Fakultas Matematika dan Ilmu Pengetahuan Alam (FMIPA) Institut Teknologi Sepuluh Nopember (ITS) Surabaya.

Dalam penulisan karya ilmiah ini penulis mendapatkan banyak pengetahuan, inspirasi, saran dan pelajaran berharga sehingga penulis menyampaikan terima kasih kepada:

1. Bapak Mardi, Ibu Yeni, Alvianti, Priscilla, Edo, Elisca dan seluruh keluarga yang selalu memberikan motivasi, doa tulus, serta bantuan baik moril maupun materi, sehingga penulis dapat menyelesaikan tesis ini tepat pada waktunya.
2. Bapak Dr. Sutikno, S.Si., M.Si. sebagai pembimbing pertama dan Bapak Dr. Drs. Agus Suharsono, M.S. sebagai pembimbing kedua yang tidak hentinya dengan sabar memberikan arahan, bimbingan, pengetahuan, dan selalu mendampingi penulis dari awal sampai akhir penulisan tesis ini.
3. Bapak Dr. I Nyoman Latra, MS. dan Ibu Dr. Ismaini Zain, M.Si. sebagai tim dosen penguji yang telah berkenan untuk menguji dan memberikan banyak ilmu pengetahuan, saran dan koreksi dalam proses penulisan tesis ini.
4. Bapak Dr. Purhadi, M.Sc. selaku dosen wali, yang menjadi pengganti orang tua yang baik selama penulis menjalani perkuliahan. Terima kasih atas bimbingan dan nasihat yang telah diberikan selama ini.
5. Seluruh tim dosen di Program Magister Jurusan Statistika FMIPA ITS Surabaya yang telah memberikan pengajaran dan ilmu pengetahuan selama penulis menjalani perkuliahan di Program Studi Statistika.

6. Seluruh tim staff di Program Magister Jurusan Statistika FMIPA ITS Surabaya yang telah membantu penulis dalam proses administrasi, dan juga kelancaran seminar proposal, seminar hasil, serta sidang dalam proses penulisan tesis.
7. Pihak Balai Penelitian Jeruk dan Buah Subtropika (Balitjestro) yang telah memberikan kesempatan bagi penulis untuk menggunakan kasus serta data penelitiannya sebagai studi kasus pada tesis ini.
8. Teman-teman AT 2011, OS, dan Stefie. yang telah banyak memberikan dukungan, motivasi, perhatian dan semangat bagi penulis untuk menyelesaikan studi di Program Magister Jurusan Statistika FMIPA ITS Surabaya.
9. Amanda, Nisa, Agnes, Riza, dan teman-teman angkatan 2015 lainnya sebagai teman seperjuangan yang telah bersama-sama melewati setiap tahapan kuliah hingga tesis, *sharing* ilmu dan keluh kesah selama menjalani perkuliahan di Program Magister Jurusan Statistika FMIPA ITS. Terimakasih telah menjadi bagian selama masa perkuliahan di Program Magister Jurusan Statistika FMIPA ITS Surabaya.
10. Terima kasih kepada pihak-pihak lain yang tidak dapat disebutkan satu per satu yang telah membantu penulis dalam menyelesaikan Tesis ini.

Penulis menyadari masih banyak kekurangan dalam penyusunan draft tesis ini. Untuk itu penulis mengharapkan kritik dan saran yang membangun. Akhir kata, penulis berharap semoga draft tesis ini dapat memberikan manfaat kepada berbagai pihak sebagaimana tujuan penulis.

Surabaya, Januari 2017

Penulis,

Alvionita

DAFTAR ISI

	Halaman
LEMBAR PENGESAHAN.....	v
ABSTRAK	vii
ABSTRACT	ix
KATA PENGANTAR	xi
DAFTAR ISI	xiii
DAFTAR GAMBAR	xvii
DAFTAR TABEL	xix
DAFTAR LAMPIRAN.....	xxi
BAB I PENDAHULUAN	
1.1 Latar Belakang	1
1.2 Rumusan Masalah	4
1.3 Tujuan Penelitian.....	4
1.4 Manfaat.....	5
1.5 Batasan Masalah.....	5
BAB II TINJAUAN PUSTAKA	
2.1 Data Numerik dan Data Kategorik.....	7
2.2 Analisis Multivariat	7
2.3 Analisis Kelompok (<i>Cluster Analysis</i>)	8
2.4 Metode Pengelompokan (<i>Clustering</i>).....	11
2.4.1 Pengelompokan Data Numerik	11
2.4.2 Pengelompokan Data Kategorik	15
2.5 Pengelompokan Data Campuran Numerik dan Kategorik	18
2.5.1 Pengelompokan Ensemble (<i>Ensemble Clustering</i>)	18
2.5.2 Pengelompokan SWFM.....	20
2.6 Kinerja Hasil Pengelompokan	23
2.7 Uji Rata-rata Dua atau Lebih Populasi	25
2.8 Fusi Protoplasma Antara Jeruk Siam Madu dan Satsuma Mandarin.....	27
2.9 Penelitian Lain dengan Studi Kasus yang Relevan.....	29

BAB III METODE PENELITIAN

3.1 Kajian Teoritis.....	31
3.1.1 Menyusun Algoritma dan Program Metode Ensemble ROCK.....	31
3.1.2 Menyusun Algoritma dan Program Metode Ensemble SWFM	34
3.2 Kajian Empiris	35
3.2.1 Data Penelitian	35
3.2.2 Identifikasi Variabel Penelitian.....	35
3.2.3 Tahapan Analisis Data.....	37
3.2.3.1 Mengelompokkan Aksesori Jeruk Hasil Fusi Protoplasma dengan Metode Ensemble ROCK dan Ensemble SWFM	37
3.2.3.2 Melakukan Perbandingan Antara Metode Ensemble ROCK dengan Ensemble SWFM	38

BAB IV ANALISIS DAN PEMBAHASAN

4.1 Penyusunan Algoritma dan Program Metode Ensemble ROCK	41
4.1.1 Algoritma Pengelompokan Ensemble ROCK	41
4.1.2 Algoritma Pemograman Ensemble ROCK di <i>R Project</i>	50
4.2 Penyusunan Algoritma dan Program Metode Ensemble SWFM	52
4.2.1 Algoritma Pengelompokan Ensemble SWFM	52
4.2.2 Algoritma Pemograman Ensemble SWFM di <i>R Project</i>	55
4.3 Hasil dan Analisis Pengelompokan dengan Metode Ensemble ROCK dan Ensemble SWFM	57
4.3.1 Deskripsi Karakteristik Aksesori Jeruk Hasil Fusi Protoplasma.....	57

4.3.1.1 Data Numerik	57
4.3.1.2 Data Kategorik.....	61
4.3.2 Pengelompokan untuk Data Numerik	63
4.3.3 Pengelompokan untuk Data Kategorik.....	68
4.3.4 Pengelompokan Data Campuran dengan Metode Ensembel ROCK.....	72
4.3.5 Pengelompokan Data Campuran dengan Metode Ensembel SWFM	76
4.4 Perbandingan Antara Metode Ensembel ROCK dengan Metode Ensembel SWFM.....	81
BAB V KESIMPULAN DAN SARAN	
5.1 Kesimpulan.....	85
5.2 Saran	86
DAFTAR PUSTAKA	87
LAMPIRAN	91
BIOGRAFI PENULIS	179

“Halaman ini sengaja dikosongkan”

DAFTAR GAMBAR

	Halaman
Gambar 2.1 Tahapan umum dalam melakukan pengelompokan ensemble	19
Gambar 2.2 Tahapan analisis dengan algoritma CEBMDC	19
Gambar 2.3 Tahapan analisis berdasarkan <i>similarity weight and filter method</i>	21
Gambar 3.1 Diagram alir analisis data menggunakan metode ensemble ROCK dan ensemble SWFM	39
Gambar 4.1 <i>Flowchart</i> prosedur pengelompokan pada metode ensemble ROCK	43
Gambar 4.2 <i>Flowchart</i> prosedur pengelompokan pada metode hirarki <i>agglomerative</i>	44
Gambar 4.3 <i>Flowchart</i> prosedur pengelompokan pada metode ROCK	49
Gambar 4.4 <i>Flowchart</i> prosedur pengelompokan pada metode ensemble SWFM.....	55
Gambar 4.5 Gambaran karakteristik aksesori jeruk berdasarkan (a) diameter buah, (b) tebal kulit, (c) jumlah juring, (d) tebal daging kulit, (e) diameter <i>axis</i> , (f) jumlah biji, (g) berat buah, dan (h) <i>brix</i>	60
Gambar 4.6 Gambaran karakteristik aksesori jeruk berdasarkan (a) bentuk buah, (b) permukaan kulit, (c) warna daging kulit, (d) tampak <i>axis</i> , (e) warna kulit, (f) <i>axis</i> buah, dan (g) tekstur <i>pulp</i>	62
Gambar 4.7 Rangkuman nilai <i>R-Square</i> berdasarkan jumlah kelompok untuk metode (a) <i>single linkage</i> , (b) <i>complete linkage</i> , dan (c) <i>single linkage</i>	65
Gambar 4.8 Rangkuman nilai <i>F</i> berdasarkan jumlah kelompok	65
Gambar 4.9 <i>Dendogram</i> hasil pengelompokan data numerik	66
Gambar 4.10 Nilai ratio pengelompokan data kategorik	70
Gambar 4.11 Nilai ratio pengelompokan ensemble ROCK	72

Gambar 4.12	Gambaran perbandingan karakteristik hasil pengelompokan ensembel ROCK berdasarkan (a) diameter buah, (b) tebal kulit, (c) jumlah juring, (d) tebal daging kulit, (e) diameter <i>axis</i> , (f) jumlah biji, (g) berat buah, dan (h) <i>brix</i>	75
Gambar 4.13	Nilai ratio pengelompokan ensembel SWFM.....	78
Gambar 4.14	Gambaran perbandingan karakteristik hasil pengelompokan ensembel SWFM berdasarkan (a) diameter buah, (b) tebal kulit, (c) jumlah juring, (d) tebal daging kulit, (e) diameter <i>axis</i> , (f) jumlah biji, (g) berat buah, dan (h) <i>brix</i>	80
Gambar 4.15	Tampilan <i>output</i> hasil perbandingan nilai ratio	82

DAFTAR TABEL

	Halaman
Tabel 2.1 Struktur data multivariat	8
Tabel 2.2 Tabel kontingensi data biner	10
Tabel 2.3 Ukuran jarak data biner.....	10
Tabel 2.4 Tabel transformasi nilai statistik uji <i>wilk's lambda</i> ke nilai F	26
Tabel 3.1 Variabel numerik buah jeruk.....	36
Tabel 3.2 Variabel kategorik buah jeruk	36
Tabel 4.1 Deskripsi karakteristik data numerik	59
Tabel 4.2 Nilai ratio hasil pengelompokan data numerik dengan 4 kelompok ..	66
Tabel 4.3 Hasil pengelompokan data kategorik dengan $\theta = 0,17$	71
Tabel 4.4 Hasil pengelompokan ensemble ROCK dengan $\theta = 0,27$	73
Tabel 4.5 Karakteristik variabel numerik hasil pengelompokan ensemble ROCK.....	76
Tabel 4.6 Karakteristik variabel kategorik hasil pengelompokan ensemble ROCK.....	76
Tabel 4.7 Hasil pengelompokan ensemble SWFM dengan 2 kelompok	78
Tabel 4.8 Karakteristik variabel numerik hasil pengelompokan ensemble SWFM	81
Tabel 4.9 Karakteristik variabel kategorik hasil pengelompokan ensemble SWFM	81

“Halaman ini sengaja dikosongkan”

DAFTAR LAMPIRAN

	Halaman
Lampiran 1. Data Pengamatan Variabel Numerik.....	91
Lampiran 2. Data Pengamatan Variabel Kategorik.....	92
Lampiran 3. <i>Flowchart</i> Prosedur Perhitungan Jarak <i>Euclidean</i>	93
Lampiran 4. <i>Flowchart</i> Prosedur Perhitungan <i>Sum Square Total</i> (SST).....	94
Lampiran 5. <i>Flowchart</i> Prosedur Perhitungan <i>Sum Square Within</i> (SSW)	95
Lampiran 6. <i>Flowchart</i> Prosedur Perhitungan <i>R-Square</i> dan <i>F</i>	96
Lampiran 7. <i>Flowchart</i> Prosedur Perhitungan Ratio S_W dan S_B untuk Data Numerik.....	97
Lampiran 8. <i>Flowchart</i> Prosedur Perhitungan <i>Similarity</i> pada Metode ROCK	98
Lampiran 9. <i>Flowchart</i> Prosedur Perhitungan <i>Link</i> pada Metode ROCK.....	99
Lampiran 10. <i>Flowchart</i> Prosedur Perhitungan <i>Goodness Measure</i> pada Metode ROCK	100
Lampiran 11. <i>Flowchart</i> Prosedur Perhitungan (a) Nilai n_k , (b) Nilai n_c , (c) Nilai n_{kc} dan (d) Ratio S_W dan S_B untuk data kategorik.....	101
Lampiran 12. <i>Flowchart</i> Prosedur Perhitungan <i>Similarity Weight</i>	103
Lampiran 13. <i>Flowchart</i> Prosedur Perhitungan Nilai Jarak F.....	104
Lampiran 14. <i>Syntax</i> Analisis Kelompok Hirarki <i>Agglomerative</i> untuk Data Numerik.....	105
Lampiran 15. <i>Syntax</i> Rangkuman Nilai <i>R-Square</i> Analisis Kelompok Hirarki <i>Agglomerative</i>	145
Lampiran 16. <i>Syntax</i> Rangkuman Nilai <i>F</i> Analisis Kelompok Hirarki <i>Agglomerative</i>	146
Lampiran 17. <i>Syntax</i> Nilai Ratio Analisis Kelompok Hirarki <i>Agglomerative</i> ...	148
Lampiran 18. <i>Syntax</i> Analisis Metode ROCK untuk Data Kategorik	151
Lampiran 19. <i>Syntax</i> Analisis Pengelompokan Ensemble ROCK	154
Lampiran 20. <i>Syntax</i> Analisis Pengelompokan Ensemble SWFM.....	157
Lampiran 21. <i>Syntax</i> Perbandingan Hasil Pengelompokan	160

Lampiran 22. <i>Output Software R</i> untuk Analisis Kelompok pada Data Numerik.....	161
Lampiran 23. <i>Output Software R</i> untuk Analisis Kelompok pada Metode ROCK Data Kategorik	167
Lampiran 24. <i>Output Software R</i> untuk Analisis Ensemble ROCK.....	169
Lampiran 25. <i>Output Software R</i> untuk Analisis Ensemble SWFM	172
Lampiran 26. Nilai <i>R-Square</i> dan <i>F</i> Analisis Pengelompokan Data Numerik ..	173
Lampiran 27. Tabel Nilai Ratio Analisis Pengelompokan Data Kategorik	174
Lampiran 28. Tabel Nilai Ratio Analisis Ensemble	175
Lampiran 29. <i>Output Software R</i> untuk Pengujian Normalitas Multivariat.....	176
Lampiran 30. <i>Output Software R</i> untuk Analisis Metode MANOVA.....	177

BAB I

PENDAHULUAN

1.1 Latar Belakang

Analisis kelompok (*cluster analysis*) merupakan salah satu teknik dalam analisis multivariat metode interdependen yang mereduksi (mengelompokkan) data. Tujuan utama analisis kelompok adalah untuk mengelompokkan objek-objek pengamatan menjadi beberapa kelompok berdasarkan karakteristik yang dimiliki. Dalam analisis kelompok, objek-objek yang paling dekat kesamaannya berada dalam kelompok yang sama, serta mempunyai kemiripan satu dengan yang lain (Johnson dan Wichern, 2002). Algoritma pengelompokan pada umumnya hanya terfokus pada pengelompokan objek dengan variabel yang memiliki data berskala numerik atau kontinu, sedangkan untuk data kategorik baru dikembangkan sekitar tahun 1990-an.

Permasalahan yang sering ditemui dalam analisis pengelompokan adalah data yang berskala campuran numerik dan kategorik. Metode yang seringkali dilakukan untuk pengelompokan data berskala campuran adalah dengan mentransformasi data kategorik menjadi data numerik dan sebaliknya. Dewangan, Sharma, dan Akasapu (2010) melakukan transformasi variabel kategorik ke dalam bentuk numerik, kemudian pengelompokan objek dilakukan dengan metode pengelompokan data numerik. Kelebihan metode transformasi adalah dapat mengurangi kompleksitas dalam komputasi. Akan tetapi, metode tersebut memiliki kelemahan dalam menentukan transformasi yang tepat agar tidak kehilangan banyak informasi dari *original* datanya.

Selain pengelompokan dengan metode transformasi tersebut, dikembangkan sebuah metode pengelompokan ensemble untuk data campuran oleh He, Xu dan Deng pada tahun 2005. Pengelompokan ensemble (*clustering ensemble*) adalah teknik pengelompokan untuk menggabungkan hasil pengelompokan beberapa algoritma pengelompokan untuk mendapatkan kelompok yang lebih baik (He, Xu, dan Deng, 2005a). He Xu, dan Deng (2005b) menerapkan teknik pengelompokan ensemble untuk mengelompokkan penyakit hati dan

persetujuan pengajuan kartu kredit dengan *Cluster Ensemble Based Mixed Data Clustering* (CEBMDC) dan membawa *combining* atau tahapan ensemble ke dalam algoritma pengelompokan untuk data kategorik, yaitu algoritma *Squeezer*. Algoritma *Squeezer* merupakan metode pengelompokan yang efektif digunakan untuk data yang berjumlah besar (Reddy dan Kavitha, 2010), sedangkan untuk data dengan jumlah yang relatif sedikit dapat digunakan metode ROCK (*RObust Clustering using linKs*).

Metode ROCK dikembangkan oleh Sudipto Guha, Rajeev Rastogi, dan Kyuseok Shim pada tahun 2000. Metode ROCK menggunakan konsep *link* untuk mengukur kesamaan/kedekatan antara pengamatan. Jumlah *link* antar pengamatan bergantung pada nilai *threshold* (θ) yang telah ditentukan. Nilai θ merupakan parameter untuk menyatakan adanya *link* antar pengamatan. Jika jarak antar sepasang pengamatan lebih besar dari nilai θ , maka pengamatan tersebut memiliki *link*. Kesalahan penentuan nilai θ dapat mengakibatkan kesalahan dalam pengelompokan, seperti setiap objek berada dalam kelompok yang sama atau setiap objek berada dalam kelompok yang berbeda. Guha melakukan pengelompokan dengan metode ROCK untuk memisahkan Demokrat dari Partai Republik dengan data *Congressional* Amerika Serikat yang disediakan oleh *UCI Machine Learning Repository*. Penelitian tersebut menunjukkan bahwa metode ROCK memiliki akurasi lebih baik daripada metode hirarki, dimana metode ini juga menunjukkan sifat skalabilitas yang baik.

Menurut Reddy dan Kavitha (2012), untuk mengatasi pengelompokan pada data campuran dapat digunakan metode ensemble berdasarkan *similarity weight and filter method* (SWFM). Metode ini merupakan perkembangan dari pengelompokan ensemble dengan perbedaan pada tahapan pembentukan *final cluster*. Pembentukan *final cluster* dilakukan berdasarkan konsep *similarity weight* dan *filter method*. Konsep *similarity weight* merupakan konsep yang digunakan untuk mengukur kesamaan/kedekatan antara pengamatan, sedangkan *filter method* merupakan nilai jarak yang dalam melakukan pengelompokan. Metode SWFM dikembangkan oleh M.V. Jagannatha Reddy dan B. Kavitha pada tahun 2012. Dengan menggunakan data keuangan, penelitian tersebut menyatakan bahwa

metode SWFM memberikan rata-rata kesalahan lebih rendah daripada metode *K-Prototype*. Reddy dan Kavitha (2012) juga membandingkan pengelompokan antara metode ensemble dan metode *K-Prototype*. Hasil yang diperoleh adalah metode ensemble memberikan rata-rata kesalahan lebih rendah daripada metode *K-Prototype*. Penelitian tersebut menunjukkan bahwa metode ensemble dan ensemble SWFM memberikan nilai kesalahan yang tidak jauh berbeda. Akan tetapi, penelitian tersebut tidak dilakukan perbandingan hasil pengelompokan antara kedua metode. Oleh karena itu, pada penelitian ini dilakukan perbandingan kinerja antara metode pengelompokan ensemble ROCK dan ensemble SWFM.

Metode ensemble ROCK dan ensemble SWFM digunakan untuk mengelompokkan aksesori jeruk hasil persilangan antara induk Siam Madu dan Satsuma Mandarin yang dilakukan oleh Balai Penelitian Jeruk dan Buah Subtropika (Balitjestro) di Kota Batu, Malang. Persilangan dilakukan melalui proses fusi protoplasma dengan harapan diperoleh jeruk siam unggul tanpa biji, rasa manis, mudah dikupas dan berwarna menarik (Martasari, 2014). Proses fusi protoplasma menghasilkan individu dengan keragaman yang tinggi, sehingga perlu dilakukan penyeleksian berdasarkan karakteristik morfologi buah yang dihasilkan. Karakteristik morfologi buah diamati dengan melibatkan variabel yang bersifat campuran numerik dan kategorik. Variabel morfologi buah jeruk yang diukur secara numerik (numerik) meliputi diameter buah, tinggi buah, berat buah, dan lain-lain. Variabel morfologi buah yang tidak dapat diukur secara numerik dan hanya dapat diberi skor (kategorik) meliputi warna kulit, bentuk buah, warna *pulp* dan lain-lain.

Dalam penelitian ini, pengelompokan data numerik dilakukan dengan metode hirarki *agglomerative*, sedangkan pengelompokan data kategorik dilakukan dengan metode ROCK. Setelah kedua *cluster* dari data numerik dan kategorik terbentuk, maka dilakukan penggabungan dengan metode ensemble ROCK dan ensemble SWFM, sehingga diperoleh satu *final cluster*. Dalam pengelompokan ensemble ROCK pada pembentukan *final cluster* menggunakan metode yang sama dengan pengelompokan data kategorik yaitu metode ROCK sedangkan pengelompokan berdasarkan SWFM menggunakan algoritma analisis jarak dan

algoritma seleksi (*filter*). Kinerja metode tersebut dilihat berdasarkan rasio dari nilai simpangan baku di dalam kelompok atau *within* (S_w) dengan nilai simpangan baku antar kelompok atau *between* (S_B). Bunkers dan James (1996) menyatakan kinerja suatu metode pengelompokan semakin baik jika semakin kecil rasio antara S_w dan S_B . Arti bahwa terdapat homogenitas maksimum dalam kelompok dan heterogenitas maksimum antar kelompok.

1.2 Rumusan masalah

Berdasarkan latar belakang yang telah dipaparkan, permasalahan penelitian ini adalah sebagai berikut,

1. Bagaimana algoritma pengelompokan ensemble ROCK pada data dengan variabel berskala campuran numerik dan kategorik ?
2. Bagaimana algoritma pengelompokan ensemble SWFM pada data dengan variabel berskala campuran numerik dan kategorik ?
3. Bagaimana hasil pengelompokan akses hasil fusi protoplasma dengan metode ensemble ROCK dan ensemble SWFM?
4. Bagaimana perbandingan kinerja metode ensemble ROCK dan ensemble SWFM untuk pengelompokan akses hasil fusi protoplasma ?

1.3 Tujuan Penelitian

Untuk menjawab rumusan permasalahan yang telah dipaparkan, maka tujuan yang ingin dicapai dari penelitian ini adalah sebagai berikut,

1. Menyusun algoritma pengelompokan ensemble ROCK pada data dengan variabel berskala campuran numerik dan kategorik.
2. Menyusun algoritma pengelompokan ensemble SWFM pada data dengan variabel berskala campuran numerik dan kategorik.
3. Mengelompokkan akses hasil fusi protoplasma dengan metode ensemble ROCK dan ensemble SWFM.
4. Membandingkan kinerja metode ensemble ROCK dan ensemble SWFM untuk pengelompokan akses hasil fusi protoplasma.

1.4 Manfaat

Penelitian ini diharapkan bermanfaat bagi,

1. Manfaat bagi Balitjestro

Hasil penelitian ini dapat digunakan untuk melakukan penyeleksian tanaman jeruk yang lebih efisien.

2. Manfaat bagi ilmu pengetahuan

Penelitian ini diharapkan dapat mengembangkan wawasan keilmuan statistika terutama dalam melakukan pengelompokan menggunakan metode ensembel ROCK dan ensembel SWFM, serta penerapannya dalam bidang pertanian.

1.5 Batasan Masalah

Penelitian ini membatasi bahwa pengelompokan aksesori jeruk dilakukan berdasarkan variabel-variabel yang ditentukan pihak Balitjestro. Metode yang digunakan untuk pengelompokan aksesori pada data numerik adalah metode pengelompokan hirarki *agglomerative* dengan jarak *euclidean*. Untuk merevisi jarak menggunakan *single linkage*, *complete linkage* dan *average linkage*. Metode yang digunakan untuk mengelompokkan aksesori pada data kategorik adalah ROCK. Kriteria perbandingan kinerja antar metode menggunakan rasio antara nilai simpangan baku di dalam kelompok dan nilai simpangan baku antar kelompok.

“Halaman ini sengaja dikosongkan”

BAB II

TINJAUAN PUSTAKA

Pada bab ini dijelaskan mengenai teori-teori yang digunakan dalam melakukan analisis pengelompokan untuk menyelesaikan studi kasus yang merupakan data multivariat berskala campuran numerik dan kategorik. Data berskala campuran tersebut tidak dapat dilakukan analisis secara parsial dikarenakan adanya hubungan antara variabel, sehingga analisis harus dilakukan secara serentak atau simultan untuk data numerik dan kategorik. Pengelompokan untuk data campuran menggunakan metode ensemble ROCK dan ensemble SWFM. Dalam analisisnya pengelompokan variabel yang berskala numerik menggunakan metode hirarki *agglomerative*, sedangkan variabel yang berskala kategorik dikelompokkan menggunakan metode ROCK.

2.1 Data Numerik dan Data Kategorik

Data merupakan fakta yang diolah menjadi suatu informasi. Menurut Anderson dan Sclove (1974), data yang digunakan dalam analisis statistik (data statistik) berdasarkan jenis variabelnya dikelompokkan menjadi dua, yaitu data numerik dan data kategorik. Data numerik merupakan data dengan variabel kuantitatif yang menghasilkan informasi numerik. Data numerik dapat dikelompokkan menjadi dua yaitu data diskrit (hasil pencacahan), misalnya jumlah anak dalam keluarga dan data kontinu (hasil pengukuran), misalnya tinggi dan berat badan. Data kategorik merupakan data dengan variabel kualitatif yang dihasilkan dari pengklasifikasian atau pengolongan suatu data (data atribut). Agresti (2007) menyatakan bahwa data kategorik memiliki skala pengukuran yang terdiri atas satu set kategori, misalnya filsafat politik yang dapat diukur sebagai kategori liberal, kategori moderat, atau kategori konservatif.

2.2 Analisis Multivariat

Analisis yang dilakukan pada data dapat berupa analisis univariat, analisis bivariat dan analisis multivariat. Menurut Johnson dan Wichern, (2007), analisis

univariat adalah analisa yang dilakukan untuk menganalisis satu variabel dari hasil pengamatan atau masing-masing variabel terhadap hasil pengamatan, sedangkan analisis multivariat adalah analisis yang berkaitan dengan jumlah variabel lebih dari dua yang dianalisis secara simultan pada masing-masing pengamatan.

Data yang digunakan dalam analisis ini adalah data yang dikumpulkan dari dua atau lebih pengamatan yang diukur oleh lebih dari satu karakteristik secara simultan atau multivariat. Suatu penelitian dengan pengamatan sebanyak n dan variabel sebanyak m memiliki struktur data multivariat seperti ditunjukkan pada Tabel 2.1.

Tabel 2.1 Struktur data multivariat

Pengamatan	Variabel 1	Variabel 2	Variabel 3	...	Variabel m
ke-i	(X_1)	(X_2)	(X_3)		(X_m)
1	x_{11}	x_{21}	x_{31}	...	x_{m1}
2	x_{12}	x_{22}	x_{32}	...	x_{m2}
3	x_{13}	x_{23}	x_{33}	...	x_{m3}
\vdots	\vdots	\vdots	\vdots	\ddots	\vdots
n	x_{1n}	x_{2n}	x_{3n}	...	x_{mn}

2.3 Analisis Kelompok (Cluster Analysis)

Metode klasifikasi merupakan bagian dari analisis statistika. Dua hal dalam pemodelan statistik untuk permasalahan klasifikasi, yaitu statistik tradisional dan modern. Pertama, statistik tradisional dikembangkan oleh Fisher di tahun 1936 dikenal dengan diskriminan linier Fisher. Kedua, statistik modern, yang memanfaatkan fleksibilitas model, menduga suatu distribusi di dalam masing-masing kelas yang pada akhirnya menyediakan suatu aturan pengelompokan (Sharma, 1996).

Analisis kelompok (*cluster analysis*) merupakan salah satu teknik dalam analisis multivariat metode interdependen yang mereduksi (mengelompokkan) data. Analisis ini mempunyai tujuan utama untuk mengelompokkan objek-objek pengamatan menjadi beberapa kelompok berdasarkan karakteristik yang dimilikinya. Analisis kelompok mengelompokkan objek-objek sehingga setiap

objek yang paling dekat kesamaannya dengan objek lain berada dalam kelompok yang sama, serta mempunyai kemiripan satu dengan yang lain (Johnson dan Wichern, 2007). Beberapa manfaat analisis kelompok adalah eksplorasi data variabel ganda, reduksi data, dan prediksi keadaan objek. Hasil analisis kelompok dipengaruhi oleh objek yang dikelompokkan, variabel yang diamati, ukuran kemiripan atau ketidakmiripan yang digunakan, skala ukuran yang digunakan, serta metode pengelompokan yang digunakan.

Ukuran kemiripan dan ketidakmiripan merupakan hal yang sangat mendasar dalam kelompok analisis. Algoritma pengelompokan menggunakan ukuran kemiripan atau ketidakmiripan digunakan untuk menggabungkan atau memisahkan data objek dari suatu data. Ukuran kemiripan biasanya digunakan oleh algoritma pengelompokan untuk menganalisis data kategori, sedangkan ukuran ketidakmiripan digunakan oleh algoritma pengelompokan untuk menganalisis data numerik. Ukuran ketidakmiripan antara objek ke- i dengan objek ke- j (d_{ij}), merupakan fungsi yang memiliki sifat-sifat sebagai berikut $d_{ij} \geq 0$, $d_{ii} = 0$, $d_{ij} = d_{ji}$, dan $d_{ik} + d_{jk} \geq d_{ij}$, untuk setiap i, j dan k . Semakin besar nilai ukuran ketidakmiripan antara dua objek maka semakin besar pula perbedaan antara kedua objek tersebut, sehingga makin cenderung untuk tidak berada dalam kelompok yang sama. (Johnson dan Wichern, 2007)

Ukuran kemiripan dan ketidakmiripan pada umumnya diukur berdasarkan jarak. Salah satu faktor yang sangat berpengaruh terhadap hasil dari kelompok yang dibentuk adalah jarak antar objek pengamatan (Sharma, 1996). Oleh karena itu, dibutuhkan suatu alat ukur untuk menentukan jarak antar objek pengamatan. Berikut ini merupakan metode-metode pengukuran jarak antara objek ke- i (x_i) dengan objek ke- j (x_j) berdasarkan karakteristik variabel yang dikelompokkan.

a. Metode pengukuran jarak untuk variabel kategorik biner

Bila variabel yang diamati berupa variabel biner yang hanya memiliki dua macam karakter yang berbeda (0,1), maka variabel yang diamati dapat dibentuk suatu tabel kontingensi seperti ditunjukkan pada Tabel 2.2. Perhitungan ukuran jarak antara variabel x_i dan x_j untuk pengukuran data biner dapat menggunakan beberapa ukuran yang disajikan pada Tabel 2.3.

Tabel 2.2 Tabel kontingensi data biner

Kategori x_i	Kategori x_j		Total
	1	0	
1	a	b	a+b
0	c	d	c+d
Total	a+c	b+d	a+b+c+d

Tabel 2.3 Ukuran jarak data biner

Jenis	Rumus
<i>Russel and Rao</i>	$RR(x_i, x_j) = \frac{a}{a+b+c+d}$
<i>Simple Matching</i>	$SM(x_i, x_j) = \frac{a+d}{a+b+c+d}$
<i>Jaccard</i>	$JACCARD(x_i, x_j) = \frac{a}{a+b+c}$
<i>Dice Czekanowski, Sørensen</i>	$DICE(x_i, x_j) = \frac{2a}{2a+b+c}$

- b. Metode pengukuran jarak untuk variabel kategorik nominal

Pada pengamatan dengan variabel nominal maka pengukuran memiliki konsep yang sama dengan *simple matcing coefficient* maupun *dice*, dimana kategorinya dapat lebih dari dua macam. Dengan jumlah variabel sebanyak m , maka rumus untuk pengukuran jarak variabel nominal antara x_i dan x_j ditunjukkan pada persamaan 2.1,

$$sim(x_i, x_j) = \frac{1}{m} \sum_{l=1}^m S_{ijl} \quad (2.1)$$

dimana $S_{ijl} = 1$ jika $x_{il} = x_{jl}$ dan $S_{ijl} = 0$ jika $x_{il} \neq x_{jl}$

- c. Metode pengukuran jarak untuk variabel kategorik ordinal

Pada pengamatan dengan variabel ordinal maka pengukuran memiliki konsep yang digunakan sama dengan metode untuk data numerik, dimana kategorinya dinyatakan sebagai suatu bilangan bulat. Salah satu metode yang dapat digunakan untuk variabel ordinal adalah jarak *manhattan*. Dengan jumlah variabel sebanyak m , maka rumus untuk pengukuran jarak x_i dan x_j pada variabel nominal ditunjukkan pada persamaan 2.2

$$sim(x_i, x_j) = \sum_{l=1}^m |x_{il} - x_{jl}| \quad (2.2)$$

d. Metode pengukuran jarak untuk variabel numerik

Pada variabel yang memiliki jenis data numerik maka jarak yang dapat digunakan adalah jarak *euclidean*. Misalkan terdapat dua observasi dengan variabel-variabel berdimensi m yaitu $\mathbf{x}_i = [x_1, x_2, \dots, x_m]^T$ dan $\mathbf{x}_j = [x_1, x_2, \dots, x_m]^T$. Konsep jarak *euclidean* yang mengukur jarak antara observasi \mathbf{x}_i dan \mathbf{x}_j adalah sebagai berikut,

$$d_{ij} = \sqrt{(\mathbf{x}_i - \mathbf{x}_j)^T (\mathbf{x}_i - \mathbf{x}_j)} \quad (2.3)$$

Terkait dengan pengertian dan tujuan dilakukannya analisis kelompok, dapat dinyatakan bahwa suatu kelompok (kelompok) yang baik adalah kelompok yang mempunyai, (Hair, Black, Babin, dan Anderson, 2009)

- a. Homogenitas (kesamaan) yang tinggi antara anggota dalam satu kelompok (*within-cluster*),
- b. Heterogenitas (perbedaan) yang tinggi antara kelompok yang satu dengan kelompok yang lain (*between cluster*).

2.4 Metode Pengelompokan (Clustering)

Tahap pengelompokan dalam analisis kelompok dibedakan menurut jenis data yang dimiliki. Pada umumnya analisis kelompok terfokus pada data numerik, akan tetapi terdapat kasus dengan data kategorik bahkan terdapat kasus dengan campuran data numerik dan kategorik. Analisis kelompok pada data kategorik tidak dapat diperlakukan seperti pada data numerik. Hal tersebut dikarenakan sifat khusus data kategorik, sehingga pengelompokan data kategorik menjadi lebih rumit dibandingkan pengelompokan untuk data numerik. (Hair, *et.al*, 2009)

2.4.1 Pengelompokan Data Numerik

Pengelompokan data numerik dilakukan berdasarkan ukuran ketakmiripan atau jarak untuk data numerik. Hasil pengelompokan disajikan dalam bentuk

dendrogram (diagram pohon) yang memungkinkan penelusuran objek-objek yang diamati menjadi lebih mudah dan informatif. Teknik yang digunakan untuk pengelompokan meliputi metode hirarki dan metode non hirarki.

Pengelompokan hirarki dimulai dengan dua atau lebih objek yang mempunyai kesamaan paling dekat, kemudian proses diteruskan ke objek lain yang mempunyai kedekatan kedua. Analisis dilakukan hingga kelompok membentuk semacam “pohon”, di mana ada hirarki (tingkatan) yang jelas antar objek, dari yang paling mirip sampai paling tidak mirip. *Dendrogram* umumnya digunakan untuk membantu memperjelas proses hirarki tersebut (Hair, *et.al*, 2009). Terdapat dua teknik pengelompokan dalam analisis kelompok hirarki yaitu teknik pembagian (*divisive*) dan teknik penggabungan (*agglomerative*).

Teknik pembagian bermula dari satu kelompok yang berunsurkan semua objek yang ada. Kelompok ini kemudian dibagi menjadi dua kelompok, dan kemudian masing-masing kelompok dibagi lagi menjadi dua kelompok, dan kemudian masing-masing kelompok dibagi lagi menjadi dua kelompok, dan seterusnya. Berbeda dengan teknik pembagian, dalam teknik penggabungan setiap objek merupakan satu kelompok tersendiri. Lalu dua kelompok yang terdekat digabungkan dan seterusnya sehingga diperoleh satu kelompok yang berunsurkan semua objek. Untuk menggabungkan dua kelompok pada tahap awal, dimana tiap kelompok hanya terdiri atas satu objek, diperlukan ukuran ketakmiripan antar objek. Bila suatu kelompok merupakan penggabungan dari beberapa kelompok sebelumnya, maka diperlukan ukuran ketakmiripan antar kelompok, kelompok-kelompok dengan ukuran ketakmiripan terkecil digabungkan menjadi kelompok yang baru.

Andaikan d_{uv} merupakan ukuran ketakmiripan antara kelompok ke- u dengan kelompok ke- v dan $d_{w(u,v)}$ merupakan ukuran ketakmiripan antara kelompok ke- w dengan kelompok (u,v) yang merupakan penggabungan antara kelompok ke- u dengan kelompok ke- v , maka beberapa teknik pengelompokan antara kelompok dinyatakan sebagai berikut, (Johnson dan Wichern, 2007)

- a. Pautan Tunggal (*Single Linkage/Nearest Neighbor*), prosedur ini didasarkan pada jarak terkecil atau jarak terdekat. Jika dua objek terpisah oleh jarak

yang pendek maka kedua objek tersebut digabung menjadi satu kelompok daan demikian seterusnya. Ukuran jarak yang digunakan adalah

$$d_{w(u,v)} = \min (d_{wu}, d_{wv}) \quad (2.4)$$

- b. Pautan Lengkap (*Complete Linkage/Farthest Neighbor*), berlawanan dengan *single linkage* prosedur ini pengelompokannya berdasarkan jarak terbesar atau jarak terjauh. Ukuran jarak yang digunakan adalah

$$d_{w(u,v)} = \max (d_{wu}, d_{wv}) \quad (2.5)$$

- c. Pautan Rataan (*Average Linkage Between Method/BAVERAGE*), prosedur ini hampir sama dengan *single linkage* maupun *complete linkage*, namun kriteria yang digunakan adalah rata-rata jarak seluruh individu dalam suatu kelompok dengan jarak seluruh individu dalam kelompok yang lain. Dengan n_u dan n_v merupakan jumlah pengamatan dalam kelompok ke- u dan ke- v , ukuran jarak yang digunakan adalah

$$d_{w(u,v)} = \frac{n_u}{n_u + n_v} d_{wu} + \frac{n_v}{n_u + n_v} d_{wv} \quad (2.6)$$

Keuntungan penggunaan metode hirarki dalam analisis kelompok adalah mempercepat pengolahan dan menghemat waktu karena data yang diinputkan membentuk hirarki atau membentuk tingkatan tersendiri sehingga mempermudah dalam penafsiran, namun kelemahan dari metode ini adalah seringnya terdapat kesalahan pada data *outlier*, perbedaan ukuran jarak yang digunakan, dan terdapatnya variabel yang tidak relevan.

Menentukan jumlah kelompok optimum merupakan tahapan penting setelah proses pengelompokan. Tahapan ini disebut sebagai validasi pengelompokan (Halkidi, Batistakis, dan Vazirgiannis, 2001). Indeks *R-Squared* merupakan salah satu indeks yang dapat digunakan untuk menentukan jumlah kelompok optimum pada pengelompokan hirarki (Sharma, 1996). Indeks tersebut melibatkan penghitungan keragaman data baik keragaman total, keragaman dalam kelompok maupun keragaman antar kelompok. Indeks validitas untuk menentukan jumlah kelompok optimum pada pengelompokan hirarki dapat dituliskan sebagai berikut,

Sum of Square Total (SST):

$$SST = \sum_{l=1}^{m_{\text{numerik}}} \sum_{i=1}^n (x_{il} - \bar{x}_l)^2 \quad (2.7)$$

Sum of Square Within Group (SSW):

$$SSW = \sum_{c=1}^C \sum_{l=1}^{m_{\text{numerik}}} \sum_{i=1}^{n_c} (x_{ilc} - \bar{x}_{lc})^2 \quad (2.8)$$

Sum of Square Between Group (SSB):

$$SSB = SST - SSW \quad (2.9)$$

dengan,

m_{numerik} : Jumlah variabel numerik dalam pengamatan,

C : Jumlah kelompok yang dibentuk dalam pengamatan,

n : Total jumlah objek pengamatan,

n_c : Jumlah anggota pada kelompok ke- c untuk $c = 1, 2, \dots, C$,

\bar{x}_l : Rata-rata keseluruhan objek pada variabel ke- l untuk $l = 1, 2, \dots, m_{\text{numerik}}$,

\bar{x}_{lc} : Rata-rata variabel ke- l pada kelompok ke- c untuk $c = 1, 2, \dots, C$.

R-Squared kelompok baru adalah rasio dari *SSB* dan *SST*. *SSB* adalah ukuran perbedaan antar kelompok, dimana $SST = SSB + SSW$. Semakin besar variasi antar kelompok maka semakin homogen variasi di dalam masing-masing kelompok, dan sebaliknya. Dari konsep diatas maka *R-Squared* dapat didefinisikan sebagai ukuran perbedaan antar kelompok, dengan nilai berkisaran dari 0 sampai 1. Nilai *R-Squared* = 0 menunjukkan bahwa tidak terdapat perbedaan antara kelompok, sedangkan nilai *R-Squared* = 1 menunjukkan adanya perbedaan yang signifikan antar kelompok yang terbentuk. Nilai *R-Squared* dirumuskan pada persamaan (2.10).

$$R\text{-Squared} = \frac{SSB}{SST} = \frac{[SST - SSW]}{SST} \quad (2.10)$$

Penentuan jumlah kelompok yang terbentuk dapat dilihat dari plot nilai indeks *R-Squared*. Jika terdapat kenaikan nilai indeks *R-Squared* yang sangat tajam pada suatu kelompok, maka pada titik tersebut merupakan jumlah kelompok yang

optimum. Selain nilai *R-Squared*, penentuan jumlah kelompok optimum dapat dilakukan dengan melihat nilai *F* atau perbandingan antara *SSB* dan *SSW* yang tertinggi.

Berbeda dengan metode hirarki, metode non hirarki dimulai dengan terlebih dahulu jumlah kelompok yang diinginkan (2 kelompok, 3 kelompok atau yang lain). Setelah jumlah kelompok diketahui, selanjutnya dilakukan proses pengelompokan. Salah satu contoh metode non hirarki adalah *K-Means Cluster*. (Hair, *et.al*, 2009). Metode non hirarki memiliki keuntungan yaitu efisiensi dalam melakukan analisis sampel dalam ukuran yang lebih besar, serta hanya memiliki sedikit kelemahan pada data *outlier* dan ukuran jarak. Kelemahan metode tersebut adalah titik yang dihasilkan acak. Metode pengelompokan hirarki dan non hirarki merupakan metode yang saling melengkapi dalam penyelesaian permasalahan. Akan tetapi, jika dalam pengelompokan objek tidak terdapat informasi apriori tentang kelompok yang terbentuk, maka metode yang digunakan metode hirarki. Dengan kata lain, hasil dari metode hirarki dapat digunakan sebagai informasi awal untuk pengelompokan objek pada metode non hirarki.

2.4.2 Pengelompokan Data Kategorik

Pengelompokan data kategorik dilakukan dengan menggunakan ukuran kemiripan atau jarak untuk data kategorik kemudian dapat dilakukan pengelompokan dengan menggunakan metode hirarki maupun non hirarki. Metode pengelompokan hirarki dan non hirarki tradisional dinilai tidak tepat digunakan pada data kategorik sehingga dikembangkan metode ROCK untuk pengelompokan data kategorik tersebut. (Guha, Rastogi, dan Shin, 2000)

Metode ROCK dikembangkan dari metode pengelompokan hirarki *agglomerative* yang digunakan untuk data kategorik. Algoritma pengelompokan hirarki tradisional yang menggunakan jarak antara titik untuk pengelompokan dinilai kurang sesuai untuk data atribut dan kategorik, sehingga dibentuk sebuah konsep baru yaitu *link* untuk mengukur kesamaan/kedekatan antara sepasang titik data. (Guha, *et.al*, 2000). Pengamatan yang mempunyai tingkat hubungan (*link*) tinggi digabungkan ke dalam satu kelompok, sedang yang mempunyai tingkat hubungan (*link*) yang kecil dipisahkan dari kelompok dimana data tersebut

dikelompokkan. Metode ROCK dapat menangani *outlier* dengan cukup efektif. Pemangkasan *outlier* memungkinkan untuk membuang yang tidak ada tetangga, sehingga titik tersebut tidak berpartisipasi dalam pengelompokan. Namun dalam beberapa situasi, *outlier* dapat hadir sebagai kelompok-kelompok kecil (Guha, *et.al*, 2000).

Pengelompokan data kategorik dengan algoritma ROCK dilakukan dengan tiga langkah. Langkah pertama adalah menghitung similaritas. Ukuran kemiripan antara pasangan objek ke- i dan objek ke- j dihitung dengan rumusan yang didefinisikan pada persamaan 2.11.

$$sim(X_i, X_j) = \frac{|X_i \cap X_j|}{|X_i \cup X_j|}, \quad i \neq j \quad (2.11)$$

dengan,

$i = 1, 2, 3, \dots, n$ dan $j = 1, 2, 3, \dots, n$

X_i : Himpunan pengamatan ke- i dengan $X_i = \{x_{1i}, x_{2i}, x_{3i}, \dots, x_{m_{kategorik\ i}}\}$,

X_j : Himpunan pengamatan ke- j dengan $X_j = \{x_{1j}, x_{2j}, x_{3j}, \dots, x_{m_{kategorik\ j}}\}$,

$|X|$: Bilangan kardinal atau jumlah anggota dari himpunan X ,

Langkah kedua adalah menentukan tetangga, pengamatan dinyatakan sebagai tetangga jika nilai $sim(X_i, X_j) \geq \theta$. Langkah terakhir adalah menghitung *link* antar objek pengamatan. Besarnya *link* dipengaruhi oleh nilai *threshold* (θ) yang merupakan parameter yang ditentukan oleh pengguna yang dapat digunakan untuk mengontrol seberapa dekat hubungan antara objek. Besarnya nilai θ yang diinputkan adalah $0 < \theta < 1$. Algoritma ROCK berhenti ketika jumlah dari kelompok yang diharapkan sudah terpenuhi atau tidak ada lagi *link* antara kelompok-kelompok. (Dutta, 2005)

Metode ROCK menggunakan informasi tentang *link* sebagai ukuran kemiripan antar objek. Jika terdapat objek pengamatan X_i , X_j , dan X_k dimana X_i tetangga dari X_j , dan X_j tetangga dari X_k , maka dikatakan X_i memiliki *link* dengan X_k walaupun X_i bukan tetangga dari X_k . Cara untuk menghitung *link* untuk semua

kemungkinan pasangan dari n objek dapat menggunakan matriks A . Matriks A merupakan matriks berukuran $n \times n$ yang bernilai 1 jika X_i dan X_j dinyatakan mirip (tetangga) dan bernilai 0 dan jika X_i dan X_j tidak mirip (bukan tetangga). Jumlah *link* antar pasangan X_i dan X_j diperoleh dari hasil kali antara baris ke X_i dan kolom ke X_j dari matriks A . Jika *link* antara X_i dan X_j semakin besar maka semakin besar kemungkinan X_i dan X_j berada dalam satu kelompok yang sama (Dutta, Mahanta, dan Arun, 2005).

Penggabungan kelompok dengan menggunakan algoritma ROCK didasarkan atas ukuran kebaikan (*goodness measure*) antar kelompok dengan rumusan pada persamaan 2.12. *Goodness measure* adalah persamaan yang menghitung jumlah *link* dibagi dengan kemungkinan *link* yang terbentuk berdasarkan ukuran kelompoknya (Tyagi dan Sharma, 2012).

$$g(C_i, C_j) = \frac{\text{link}[C_i, C_j]}{(n_i + n_j)^{1+2f(\theta)} - n_i^{1+2f(\theta)} - n_j^{1+2f(\theta)}} \quad (2.12)$$

dengan $\text{link}[C_i, C_j] = \sum_{X_i \in C_i, X_j \in C_j} \text{link}(X_i, X_j)$ yang menyatakan jumlah *link* dari

semua kemungkinan pasangan objek yang ada dalam C_i dan C_j , serta n_i dan n_j masing-masing menyatakan jumlah anggota dalam kelompok ke i dan j , sedangkan

$$f(\theta) = \frac{1-\theta}{1+\theta}.$$

Penelitian pertama mengenai metode ROCK dilakukan oleh Sudito Guha, Rejeev Rastogi, dan Kyuseok Shim dengan melakukan teknik ROCK untuk memisahkan demokrat dari partai republik dengan menggunakan data *Congressional Amerika Serikat* yang disediakan oleh *UCI Machine Learning Repository*. Hasil yang diperoleh adalah kelompok untuk republik ditemukan oleh algoritma tradisional, sekitar 25% dari anggota adalah demokrat, sementara dengan ROCK, hanya 12% yang demokrat. Hal ini menunjukkan bahwa hasil pengelompokan dengan metode ROCK lebih baik daripada algoritma tradisional.

Mulyono (2006) melakukan pengelompokan dengan metode ROCK pada tiga jenis data berbeda. Dalam penelitian tersebut diperoleh bahwa pada data I dengan menggunakan algoritma hirarki tradisional pengelompokan dijadikan menjadi dua kelompok, akan tetapi bila menggunakan algoritma ROCK dengan

$\theta = 0,5$ hanya bisa dijadikan paling kecil dua kelompok. Data II dengan $\theta = 0,42$ paling kecil didapatkan tiga kelompok. Data III dengan menggunakan $\theta = 0,7$ dan $k = 3$ pengelompokan didapatkan tiga kelompok. Kesimpulannya adalah hasil dari metode ROCK tidak hanya ditentukan oleh besarnya nilai k , akan tetapi juga memperhatikan *link* antar kelompok.

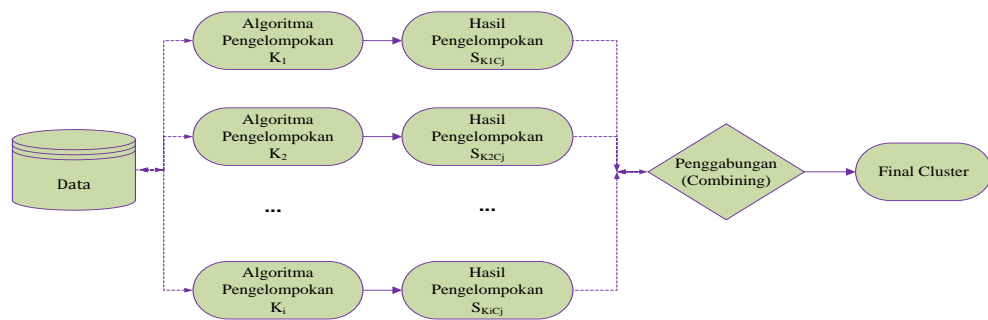
2.5 Pengelompokan Data Campuran Numerik dan Kategorik

Pengelompokan data campuran numerik dan kategorik dilakukan dengan membagi data tersebut menjadi dua sub-data, yaitu murni numerik dan murni kategorik. Misalkan terdapat data dengan variabel berskala campuran sebanyak m , dengan $m_{numerik}$ merupakan jumlah variabel numerik, dan $m_{kategorik}$ merupakan jumlah variabel kategorik, sehingga diperoleh $m = m_{numerik} + m_{kategorik}$. Selanjutnya dilakukan pengelompokan objek dengan algoritma pengelompokan data sesuai jenis data secara terpisah. Hasil pengelompokan tersebut digabungkan menggunakan metode pengelompokan ensemble berdasarkan ROCK dan ensemble berdasarkan SWFM sehingga diperoleh kelompok akhir (*final cluster*).

2.5.1 Pengelompokan Ensemble (Ensemble Clustering)

Pengelompokan ensemble merupakan metode yang menggabungkan beberapa algoritma yang berbeda untuk mendapatkan partisi umum dari data, yang bertujuan untuk konsolidasi dari portofolio hasil pengelompokan individu (Suguna dan Selvi, 2012). Tujuan pengelompokan ensemble adalah untuk menggabungkan hasil Pengelompokan dari beberapa algoritma pengelompokan untuk mendapatkan hasil pengelompokan yang lebih baik dan *robust* (Yoon, Ahn, Lee, Cho, dan Kim, 2006).

Pengelompokan ensemble terdiri atas dua tahap algoritma. Tahap pertama adalah melakukan pengelompokan dengan beberapa algoritma dan menyimpan hasil pengelompokan tersebut. Kedua, menggunakan fungsi konsensus untuk menentukan *final cluster* dari kelompok-kelompok hasil tahap pertama. Skema pengelompokan ensemble secara umum ditunjukkan pada Gambar 2.1, sedangkan skema pengelompokan ensemble untuk data campuran dengan algoritma CEBMDC secara umum ditunjukkan pada Gambar 2.2.

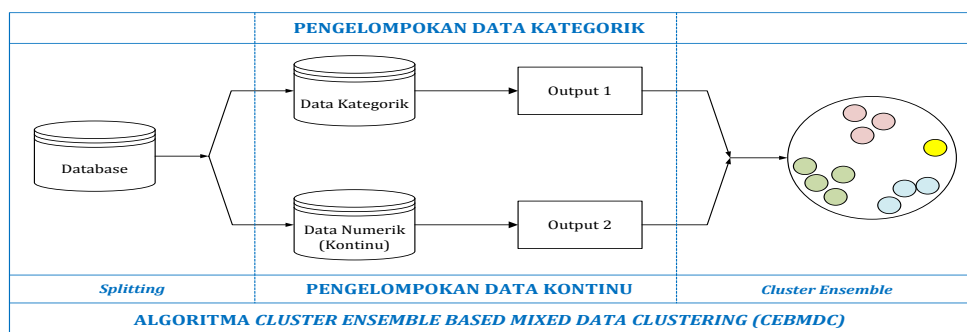


Sumber : Irfandi (2016)

Gambar 2.1 Tahapan umum dalam melakukan pengelompokan ensemble.

Langkah-langkah dalam analisis data campuran menggunakan metode pengelompokan ensemble yang disebut Algoritma CEBMDC memiliki tahapan sebagai berikut, (He, Xu, dan Deng, 2005b)

- Membagi data menjadi dua subdata, yaitu murni numerik dan murni kategorik.
- Melakukan pengelompokan objek dengan variabel numerik dengan algoritma pengelompokan data numerik, serta melakukan pengelompokan objek dengan variabel kategorik dengan algoritma pengelompokan data kategorik.
- Menggabungkan (*combining*) hasil pengelompokan dari variabel numerik dan kategorik, yang disebut proses ensemble.
- Melakukan pengelompokan ensemble menggunakan algoritma pengelompokan data kategorik untuk mendapatkan kelompok akhir (*final cluster*).



Sumber : Irfandi (2016)

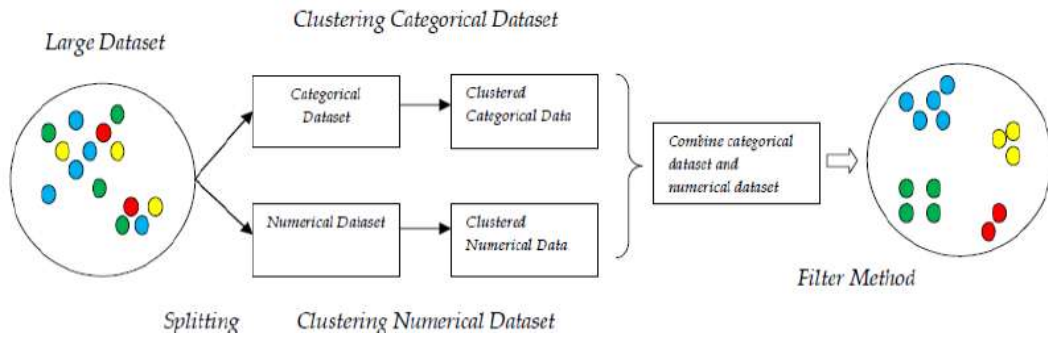
Gambar 2.2 Tahapan analisis dengan algoritma CEBMDC.

Terdapat beberapa penelitian yang menggunakan metode ensemble, salah satunya yang dilakukan oleh Dewi (2012). Penelitian yang dilakukan adalah pengelompokan desa di Provinsi Riau menggunakan metode ensemble dengan algoritma *squeezer*. Hasil pengelompokan menunjukkan bahwa metode pengelompokan ensemble menghasilkan kelompok dengan kinerja yang lebih baik daripada pengelompokan full kategorik dan full kontinu. Nilai rasio dari masing-masing metode yang diukur pada jumlah kelompok sama dengan 4 secara berturut-turut adalah 0,0072; 0,0904; dan 0,2679. Dengan pengelompokan ensemble, desa perdesaan di Provinsi Riau dapat dikelompokkan menjadi 4 kelompok.

Rahayu (2013) melakukan metode ensemble untuk pengelompokan data mahasiswa non aktif Program Studi Matematika FMIPA, Universitas Terbuka. Data kategori diproses menggunakan algoritma QROCK, menghasilkan 44 kelompok. Data numerik diproses menggunakan algoritma AGNES, menghasilkan 69 kelompok. Hasil dari kedua pengelompokan digabung kemudian diproses menggunakan algoritma QROCK. Kelompok-kelompok yang dihasilkan memiliki kesamaan karakteristik pada pendidikan akhir, status pekerjaan, status perkawinan, dan jenis kelamin. Faktor prestasi akademik menunjukkan bahwa tingkat kelulusan matakuliah dalam dua semester pertama sangat rendah. Dapat dikatakan bahwa dua semester pertama merupakan masa kritis bagi mahasiswa Program Studi Matematika UT.

2.5.2 Pengelompokan SWFM

Pengelompokan SWFM memiliki konsep analisis yang sama dengan pengelompokan ensemble pada umumnya. Metode ini merupakan perkembangan dari pengelompokan ensemble yang memiliki perbedaan pada tahapan pembentukan *final cluster*. Dalam metode ensemble, pembentukan *final cluster* dilakukan dengan menggunakan algoritma pengelompokan yang sama dengan pengelompokan pada data kategorik, sedangkan ensemble SWFM menggunakan algoritma yang berdasarkan dengan *similarity weight* dan *filter method*. Tahapan pengelompokan SWFM secara umum ditunjukkan pada Gambar 2.3.



Sumber : Reddy dan Kavitha (2012).

Gambar 2.3 Tahapan analisis berdasarkan *similarity weight and filter method*.

Pada tahapan metode *similarity weight* digunakan ukuran kemiripan yang memasukkan faktor bobot pada rumus ukuran kemiripan. Bobot yang diberikan tergantung pada jumlah anggota pengamatan (n_i atau n_j). Rumusan yang digunakan untuk menghitung ukuran kemiripan antara pasangan objek ke- i dan objek ke- j didefinisikan pada persamaan 2.13 (Reddy dan Kavitha, 2012).

$$sim(X_i, X_j) = \sum_{i \leq n_i, j \leq n_j} \frac{S_{ij}}{\max(n_i, n_j)}, \quad i \neq j \quad (2.13)$$

dengan,

$$i = 1, 2, 3, \dots, n, \quad j = 1, 2, 3, \dots, n \text{ dan } S_{ij} = \frac{|X_i \cap X_j|}{|X_i \cup X_j|}$$

X_i : Himpunan pengamatan ke- i dengan $X_i = \{x_{1i}, x_{2i}, x_{3i}, \dots, x_{m_{kategorik\ i}}\}$,

X_j : Himpunan pengamatan ke- j dengan $X_j = \{x_{1j}, x_{2j}, x_{3j}, \dots, x_{m_{kategorik\ j}}\}$,

$|X|$: Bilangan kardinal atau jumlah anggota dari himpunan X ,

$m_{kategorik}$: Jumlah variabel kategorik dalam pengamatan,

n : Total jumlah objek pengamatan

n_i : Jumlah anggota dalam kelompok ke- i ,

n_j : Jumlah anggota dalam kelompok ke- j .

Untuk mengelompokkan data campuran numerik dan kategorik campuran, diusulkan sebuah algoritma yang disebut metode filter berdasarkan *similarity weight*. Pertama, data asli dibagi menjadi dua sub-data yaitu, data kategorik murni

dan data numerik murni. Berikutnya, diterapkan pengelompokan sub-data berdasarkan jenis mereka data untuk mendapatkan yang sesuai kelompok. Terakhir, hasil pengelompokan data numerik dan kategorik digabungkan sebagai data kategorik dan dikelompokkan untuk didapatkan hasil pengelompokan akhir (*final cluster*) dengan memanfaatkan metode *filter* (Reddy dan Kavitha, 2012). Persamaan yang digunakan sebagai jarak pengelompokan dengan metode *filter* didefinisikan pada persamaan 2.14.

$$F(X_i, X_j) = \sum_{i=1}^{n_i} \sum_{j=1}^{n_j} w_{ij} d(X_i, X_j) \quad (2.14)$$

dengan $d(X_i, X_j) = 1 - \text{sim}(X_i, X_j)$ dan w_{ij} merupakan bobot antara kelompok ke i dan j yang pada umumnya bernilai 0,5.

Proses penyaringan informasi atau pola menggunakan teknik yang melibatkan kolaborasi antara beberapa sudut pandang dan sumber data. Metode *filtering* kolaboratif telah diterapkan untuk berbagai jenis data termasuk penginderaan dan pemantauan data seperti dalam eksplorasi mineral, penginderaan lingkungan di daerah yang luas atau beberapa sensor (Reddy dan Kavitha, 2012).

Beberapa keuntungan yang diperoleh dengan menggunakan metode ini adalah (1) Penggunaan yang efisien dari *cut* dan siklus properti, (2) Bentuk cluster memiliki dampak yang sangat kecil pada kinerja algoritma *clustering*, (3) Efisien untuk dimensi lebih dari 5 dan mengurangi kompleksitas waktu, (4) Pencarian tetangga terdekat digunakan lebih efisien, (5) Bekerja secara efisien bahkan jika batas-batas kelompok yang tidak teratur. (Reddy dan Kavitha, 2012)

Tidak terdapat banyak penelitian yang menggunakan metode SWFM, sehingga tidak terdapat banyak referensi mengenai metode ini. Penelitian pertama mengenai metode ini dilakukan oleh Reddy dan Kavitha (2012), penelitian tersebut menggunakan data keuangan. Hasil penelitian tersebut menyatakan bahwa metode SWFM memberikan rata-rata kesalahan sebesar 0,181. Rata-rata tersebut lebih rendah dibandingkan metode *K-Prototype* yang memberikan kesalahan sebesar 0,311. Hal ini menunjukkan bahwa metode SWFM lebih baik digunakan dibandingkan metode *K-Prototype*.

2.6 Kinerja Hasil Pengelompokan

Pengukuran kinerja hasil pengelompokan merupakan langkah untuk mengetahui validitas suatu pengelompokan. Kelompok yang baik adalah memiliki kehomogenan yang tinggi antar anggota dalam kelompok dan keheterogenan yang tinggi antar kelompok (Hair, *et.al*, 2009). Menurut Bunkers dan James (1996), kinerja hasil pengelompokan untuk variabel dengan skala data numerik dapat diketahui dari rasio nilai S_W dan S_B . Dengan menggunakan nilai rata-rata variabel, simpangan baku di dalam kelompok atau *within* (S_w) dan simpangan baku antar kelompok atau *between* (S_B) dapat dirumuskan seperti pada persamaan (2.15) dan (2.16).

$$S_W = \frac{1}{C} \sum_{c=1}^C S_c \quad (2.15)$$

dengan S_c merupakan simpangan baku kelompok ke- c dan C adalah jumlah kelompok yang terbentuk.

$$S_B = \left[\frac{1}{C-1} \sum_{c=1}^C (\bar{x}_c - \bar{x})^2 \right]^{1/2} \quad (2.16)$$

dimana \bar{x}_c adalah rata-rata kelompok ke- c dan \bar{x} adalah rata-rata keseluruhan kelompok. Kinerja suatu metode pengelompokan semakin baik, jika semakin kecil nilai rasio antara S_W dan S_B . Hal ini berarti bahwa terdapat homogenitas maksimum dalam kelompok dan heterogenitas maksimum antar kelompok (Bunkers dan James, 1996).

Pengukuran kinerja dengan rasio S_W dan S_B hanya digunakan untuk data numerik sedangkan untuk data yang kategorik adalah dengan menggunakan tabel kontingensi yang ekuivalen dengan melakukan ANOVA (*Analysis of Variance*). Ukuran keragaman untuk data kategorik dikembangkan oleh Light dan Margolin (1971), Okada (1999) serta Kader dan Perry (2007). Jika terdapat sebanyak n pengamatan dengan n_k merupakan jumlah pengamatan dengan kategori ke- k dimana $k = 1, 2, 3, \dots, K$ dan $\sum_{k=1}^K n_k = n$. Selanjutnya, n_{kc} merupakan jumlah pengamatan dengan kategori ke- k dan kelompok ke- c , dimana $c = 1, 2, 3, \dots, C$

dengan C adalah jumlah kelompok yang terbentuk, sehingga $n_{.c} = \sum_{k=1}^K n_{kc}$ merupakan

jumlah pengamatan pada kelompok ke-c dan $n_{k.} = \sum_{c=1}^C n_{kc}$ merupakan jumlah pengamatan pada kategori ke-k. Total jumlah pengamatan dapat dituliskan menjadi

$$n = \sum_{c=1}^C n_{.c} = \sum_{k=1}^K n_{k.} = \sum_{k=1}^K \sum_{c=1}^C n_{kc}.$$

Jumlah kuadrat total (*SST*) untuk sebuah variabel dengan data kategorik dapat dirumuskan seperti persamaan (2.17). Untuk total jumlah kuadrat dalam kelompok (*SSW*) dirumuskan dalam persamaan (2.18). Serta jumlah kuadrat antar kelompok (*SSB*) dapat dirumuskan seperti pada persamaan (2.19). (Dewi, 2012)

$$SST = \frac{n}{2} - \frac{1}{2n} \sum_{k=1}^K n_k^2 \quad (2.17)$$

$$SSW = \sum_{c=1}^C \left(\frac{n_{.c}}{2} - \frac{1}{2n_{.c}} \sum_{k=1}^K n_{kc}^2 \right) = \frac{n}{2} - \frac{1}{2} \sum_{c=1}^C \frac{1}{n_{.c}} \sum_{k=1}^K n_{kc}^2 \quad (2.18)$$

$$SSB = \frac{1}{2} \left(\sum_{c=1}^C \frac{1}{n_{.c}} \sum_{k=1}^K n_{kc}^2 \right) - \frac{1}{2n} \sum_{k=1}^K n_k^2. \quad (2.19)$$

Mean of squares total (MST), *mean of squares within (MSW)*, dan *mean of squares between (MSB)* dapat dirumuskan seperti pada persamaan (2.20), (2.21), dan (2.22).

$$MST = \frac{SST}{(n-1)} \quad (2.20)$$

$$MSW = \frac{SSW}{(n-C)} \quad (2.21)$$

$$MSB = \frac{SSB}{C-1} \quad (2.22)$$

Simpangan baku dalam kelompok (S_w) dan simpangan baku antar kelompok (S_B) untuk data kategorik dapat dirumuskan seperti pada persamaan (2.23) dan (2.24).

$$S_w = [MSW]^{1/2} \quad (2.23)$$

$$S_B = [MSB]^{1/2} \quad (2.24)$$

Seperti halnya dengan data numerik, kinerja suatu metode pengelompokan untuk data kategorik semakin baik jika semakin kecil rasio antara S_W dan S_B , yang berarti bahwa terdapat homogenitas maksimum dalam kelompok dan heterogenitas maksimum antar kelompok (Bunkers dan James, 1996) .

2.7 Uji Rata-rata Dua atau Lebih Populasi

Setelah diperoleh hasil pengelompokan yang optimum, berikutnya dilakukan uji perbedaan rata-rata dari kelompok diperoleh. Pada data multivariat, pengujian rata-rata tersebut dilakukan dengan menggunakan *multivariate analysis of variance* (MANOVA). Menurut Johnson dan Wichern, (2007), MANOVA merupakan perluasan dari teknik univariat *analysis of variance* (ANOVA) yang melibatkan lebih dari satu variabel (multivariat). Rumusan hipotesis pada MANOVA adalah sebagai berikut,

$$H_0 : \bar{\mu}_1 = \bar{\mu}_2 = \dots = \bar{\mu}_C = \bar{\mu}$$

atau rata-rata dari setiap kelompok adalah sama.

$$H_1 : \text{minimal terdapat satu } \bar{x}_c \neq \bar{x} \text{ untuk } c = 1, 2, \dots, C$$

atau rata-rata dari setiap kelompok adalah berbeda.

Salah satu statistik uji yang dapat digunakan dalam pengambilan keputusan dalam perbedaan antar kelompok adalah *wilk's lambda*. Nilai statistik uji *wilk's lambda* berkisar antara 0 sampai 1. Semakin rendah nilai statistik *wilk's lambda*, maka perbedaan antara kelompok semakin signifikan. Rumusan untuk statistik uji *wilk's lambda*, didefinisikan pada persamaan 2.25 (Johnson dan Wichern, 2007).

$$\Lambda^* = \frac{|\mathbf{W}|}{|\mathbf{B} + \mathbf{W}|} \quad (2.25)$$

dimana $|\mathbf{W}|$ merupakan nilai SSW dengan rumusan pada persamaan 2.8 dan $|\mathbf{W} + \mathbf{B}|$ merupakan matriks nilai SST dengan rumusan pada persamaan 2.7. Dalam pengambilan keputusan, nilai statistik uji *wilk's lambda* ditransformasi kebesaran statistik F sehingga dapat dilakukan perbandingan dengan tabel F . Bentuk transformasi nilai statistik uji *wilk's lambda* ke nilai F untuk berbagai kombinasi jumlah variabel dan derajat bebas disajikan pada Tabel 2.4

Tabel 2.4 Tabel transformasi nilai statistik uji *wilk's lambda* ke nilai *F*

Jumlah variabel ($m_{numerik}$)	Jumlah Kelompok (C)	Transformasi F
$m_{numerik} = 1$	$C \geq 2$	$\left(\frac{\sum n_c - C}{C-1} \right) \left(\frac{1 - \Lambda^*}{\Lambda^*} \right) \sim F_{C-1, \sum n_c - C}$
$m_{numerik} = 2$	$C \geq 2$	$\left(\frac{\sum n_c - C - 1}{C-1} \right) \left(\frac{1 - \sqrt{\Lambda^*}}{\sqrt{\Lambda^*}} \right) \sim F_{2(C-1), 2(\sum n_c - C - 1)}$
$m_{numerik} \geq 1$	$C = 2$	$\left(\frac{\sum n_c - m_{numerik} - 1}{m_{numerik}} \right) \left(\frac{1 - \Lambda^*}{\Lambda^*} \right) \sim F_{m_{numerik}, (\sum n_c - m_{numerik} - 1)}$
$m_{numerik} \geq 1$	$C = 3$	$\left(\frac{\sum n_c - m_{numerik} - 2}{m_{numerik}} \right) \left(\frac{1 - \sqrt{\Lambda^*}}{\sqrt{\Lambda^*}} \right) \sim F_{2m_{numerik}, 2(\sum n_c - m_{numerik} - 2)}$

Dalam melakukan analisis dengan metode MANOVA, data yang dimiliki harus berdistribusi normal multivariat. Pengujian distribusi normal multivariat secara visual dapat dilakukan dengan cara membuat Q-Q plot dari nilai jarak menggunakan pendekatan *mahalanobis*. Selain pengujian secara visual menggunakan Q-Q plot perhitungan untuk pengujian normal multivariat dapat dilihat dari pengujian hipotesis melalui nilai skewnes dan kurtosis untuk mengetahui apakah data berdistribusi normal multivariat atau tidak.

Ketika $\mathbf{x} \sim N(\boldsymbol{\mu}, \mathbf{S})$ dan nilai momen pusat pada orde ketiga untuk distribusi normal multivariat adalah nol maka diperoleh $\beta_{1, m_{numerik}} = 0$ dan $\beta_{2, m_{numerik}} = m_{numerik}(m_{numerik} + 2)$. Nilai taksiran untuk $\beta_{1, m_{numerik}}$ dan $\beta_{2, m_{numerik}}$ pada n data dan $m_{numerik}$ variabel diperoleh dari persamaan 2.28 dan 2.29 (Mardia, 1970).

$$b_{1, m_{numerik}} = \frac{1}{n^2} \sum_{i=1}^n g_i^3 \quad (2.28)$$

$$b_{2, m_{numerik}} = \frac{1}{n} \sum_{i=1}^n g_i^2 \quad (2.29)$$

$$g_i = (\mathbf{x} - \boldsymbol{\mu})' \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}) \quad (2.30)$$

Dengan nilai g_i merupakan nilai jarak *mahalanobis* yang diperoleh dengan persamaan 2.30 sehingga untuk nilai skewnes dalam pengujian normal multivariat dapat menggunakan hipotesis sebagai berikut,

$H_0 : \beta_{1,m_{numerik}} = 0$ (data berdistribusi normal multivariat).

$H_1 : \beta_{1,m_{numerik}} \neq 0$ (data tidak berdistribusi normal multivariat).

Statistik uji yang digunakan didefinisikan ada persamaan 2.31. Dengan menggunakan statistik uji tersebut maka tolak hipotesis nol (H_0) jika $z_{skew} \geq \chi^2_{\frac{1}{6}m_{numerik}(m_{numerik}+1)(m_{numerik}+2)}$ atau tolak H_0 jika nilai $p\text{-value} \leq \alpha$.

$$z_{skew} = \frac{(m_{numerik} + 1)(n+1)(n+3)}{6\{(n+1)(m_{numerik} + 1) - 6\}} b_{1,m_{numerik}} \quad (2.31)$$

Sedangkan untuk nilai kurtosis dapat dilihat apakah distribusi terlalu menanjak ataupun tidak, maka rumusan hipotesis yang digunakan adalah,

$H_0 : \beta_{2,m_{numerik}} = 0$ (data berdistribusi normal multivariat).

$H_1 : \beta_{2,m_{numerik}} \neq 0$ (data tidak berdistribusi normal multivariat).

Statistik uji yang digunakan didefinisikan ada persamaan 2.32. Dengan menggunakan statistik uji tersebut maka tolak H_0 jika nilai z_{kurt} lebih besar dari batas atas 2.5% atau lebih kecil dari nilai batas bawah 25% dari distribusi z atau tolak H_0 jika nilai $p\text{-value} \leq \alpha$.

$$z_{kurt} = \frac{b_{1,m_{numerik}} - m_{numerik} (m_{numerik} + 2)}{\sqrt{\{8m_{numerik} (m_{numerik} + 2)\} / n}} \quad (2.32)$$

2.8 Fusi Protoplasma Antara Jeruk Siam Madu dan Satsuma Mandarin

Pemuliaan antara tanaman jeruk Siam Madu dan Satsuma Mandarin telah dimulai sejak tahun 2006 oleh Balitjestro. Cara pemuliaan yang digunakan adalah metode fusi protoplasma. Hal ini dikarenakan Satsuma Mandarin (*C. unshiu Marc.*) merupakan jenis jeruk yang secara alami mempunyai sifat *seedless* dan telah terbukti bahwa sifat *seedless* yang terdapat pada jeruk Satsuma Mandarin disebabkan oleh polennya yang steril (*male sterility*) yang biasa disebut dengan istilah MS (Husni, 2010). Untuk memindahkan sifat MS tersebut dari jeruk Satsuma Mandarin kepada jeruk Siam Madu sangat sulit dilakukan melalui pemuliaan konvensional karena adanya faktor genetik (*incompatible*). Oleh karena itu, pemuliaan dilakukan dengan metode fusi protoplasma yang merupakan teknik yang

digunakan dalam kegiatan persilangan tanaman dengan cara menggabungkan dua sel tanaman yang berbeda, sehingga memperoleh tanaman hibrida. (Husni, 2010).

Hasil tanaman yang diharapkan adalah jeruk Siam unggul tanpa biji, rasa yang manis, kulit mudah dikupas dan memiliki warna yang menarik (Martasari, 2014). Rasa manis pada jeruk dapat dilihat dari beberapa karakteristik jeruk diantaranya kulit jeruk yang mulus, kulit jeruk yang tidak tebal, kadar air (*brix*) yang tinggi, bentuk buah yang agak oval, tekstur *pulp* yang lembut, dan sebagainya. Hasil dari pemuliaan antara jeruk Siam Madu dan Satsuma Mandarin adalah 120 tanaman fusan yang selanjutnya disebut aksesori, akan tetapi hanya terdapat 25 dari 120 aksesori yang dapat diamati dikarenakan 95 aksesori lainnya tidak berbuah matang atau gagal panen.

Penggabungan protoplas yang melibatkan dua atau lebih protoplas menimbulkan keragaman individu baru yang dihasilkan sangat tinggi. Hal ini menyebabkan sulitnya mengetahui aksesori yang merupakan varietas unggul, sehingga perlunya dilakukan penyeleksian terhadap individu baru hasil fusi (fusan) dengan cara melakukan pengamatan karakter morfologi buah jeruk yang dihasilkan. Penyeleksian digunakan untuk memperoleh kelompok varietas unggul dari hasil pemuliaan untuk diamati lebih lanjut. Banyaknya aksesori mengakibatkan proses seleksi semakin sulit dikarenakan antar aksesori memiliki karakteristik yang berbeda, walaupun terdapat pula aksesori yang memiliki karakteristik yang sama. Oleh karena itu untuk mempermudah seleksi dapat dilakukan menggunakan metode pengelompokan (*clustering*) terhadap aksesori sesuai dengan kemiripan (*similarity*) karakteristiknya.

Hasil dari fusi protoplasma merupakan suatu aksesori baru. Aksesori tersebut diamati berdasarkan buah jeruk yang dihasilkan. Karakteristik morfologi buah jeruk terdiri atas dua jenis variabel, yaitu variabel berskala kategorik dan variabel berskala numerik. Variabel numerik morfologi buah jeruk meliputi variabel yang dapat diukur secara numerik seperti diameter buah, tinggi buah, berat buah, dan lain-lain. Sedangkan variabel kategorik morfologi buah meliputi variabel yang tidak dapat diukur secara numerik dan hanya dapat diberi skor seperti warna kulit, bentuk buah, warna *pulp* dan lain-lain.

Karakteristik yang diamati tersebut tidak dapat dilakukan analisis secara parsial antara variabel numerik dan kategorik. Hal ini dikarenakan karakteristik tersebut saling mempengaruhi dan berasal dari satu objek penelitian yang sama, sehingga analisis harus dilakukan secara multivariat dan melibatkan seluruh variabel baik numerik dan kategorik secara serentak. Jika karakteristik ini dianalisis secara parsial, sehingga memberikan hasil yang tidak maksimal. Hasil yang diperoleh jika dilakukan analisis secara parsial adalah adanya ketidakkonsistenan anggota kelompok yang ada pada kelompok yang terbentuk, sehingga terjadinya multi kelompok dari aksesori yang sama, sehingga tidak dapat dilakukan pengelompokan yang pasti terhadap aksesori yang diamati. Hal ini yang menjadi dasar bahwa aksesori jeruk harus dikelompokkan dengan menggunakan metode pengelompokan untuk data campuran, sehingga studi kasus yang dipilih sesuai dengan metode analisis yang digunakan.

2.9 Penelitian Lain dengan Studi Kasus yang Relevan

Analisis pengelompokan terhadap tanaman jeruk berdasarkan variabel numerik dan kategorik ini sebelumnya telah dilakukan oleh Irfandi (2016). Penelitian tersebut menggunakan metode pengelompokan ensemble dengan algoritma *Squeezer*. Hasil pengelompokan aksesori jeruk hasil fusi protoplasma yang diperoleh adalah aksesori dapat dikelompokkan menjadi 5 kelompok dengan metode hirarki, 3 kelompok dengan algoritma *squeezer*, dan 3 kelompok dengan metode ensemble. Pada penelitian ini, hasil pengelompokan terbaik merupakan pengelompokan yang diperoleh dari pengelompokan metode hirarki dengan jarak *euclidean* dan pautan *average*.

Penelitian lain dilakukan oleh Saputra (2016) yang mengelompokkan tanaman jeruk dengan jenis yang berbeda menggunakan metode ensemble dengan *Fuzzy C-Means* dan *K-Modes*. Kesimpulan yang diperoleh adalah dari perbandingan tiga metode (*Fuzzy C-Means*, *K-Modes*, dan gabungan) dinyatakan bahwa metode *Fuzzy C-Means* menjadi algoritma terbaik. Hal ini dikarenakan metode *Fuzzy C-Means* memiliki nilai *icdrate* terendah dengan akurasi yang cukup tinggi. Akan tetapi, metode pengelompokan ensemble dengan *Fuzzy C-Means* dan

K-Modes merupakan perkembangan dari metode pengelompokan non hirarki sehingga kurang sesuai digunakan jika pada kasus tidak memiliki jumlah kelompok yang dibentuk.

Kedua hasil penelitian yang telah dilakukan diatas menunjukkan bahwa metode dengan transformasi ke variabel numerik memberikan hasil yang lebih baik dibandingkan metode dengan campuran variabel numerik dan kategorik. Akan tetapi, hasil dari kedua penelitian diatas dinilai kurang tepat. Hal ini dikarenakan harapan dari dilakukannya penelitian tersebut adalah untuk menunjukkan bahwa metode campuran variabel numerik dan kategorik memberikan hasil yang lebih baik dibandingkan metode satu variabel. Dengan kata lain hasil yang diperoleh tidak sesuai dengan hasil yang ditunjukkan dalam jurnal yang dijadikan tinjauan pustaka dalam penelitian tersebut.

Dengan mempertimbangkan hasil dari kedua penelitian diatas, dalam penelitian ini dilakukan analisis dengan menggunakan metode yang berbeda. Metode yang digunakan adalah metode pengelompokan ensemble ROCK dan ensemble SWFM, dimana dalam analisisnya pengelompokan data numerik digunakan metode hirarki *agglomerative* sedangkan data kategorik dikelompokkan menggunakan metode ROCK. Pemilihan metode disesuaikan dengan karakteristik data yang berukuran kecil, sehingga mampu memberikan kinerja yang lebih baik dengan melibatkan seluruh karakteristik yang ada untuk menyelesaikan studi kasus.

BAB III

METODE PENELITIAN

Pada bab ini dijelaskan mengenai urutan kerja analisis secara umum dalam menyelesaikan algoritma pengelompokan untuk menyelesaikan studi kasus yang merupakan data multivariat berskala campuran numerik dan kategorik. Metode yang digunakan adalah metode ensemble ROCK dan ensemble SWFM. Dalam analisisnya, data numerik dikelompokkan menggunakan metode hirarki *agglomerative*, sedangkan data kategorik menggunakan metode ROCK.

3.1 Kajian Teoritis

Kajian teoritis merupakan kajian yang berhubungan dengan metode yang digunakan sebagai alat analisis dalam penelitian. Kajian teoritis dilakukan untuk mencapai tujuan pertama dan kedua tujuan penelitian. Kajian teoritis yang dilakukan meliputi penyusunan algoritma dan pemograman untuk metode ensemble ROCK dan ensemble SWFM.

3.1.1 Menyusun Algoritma dan Program Metode Ensemble ROCK

Metode ensemble yang digunakan adalah algoritma CEBMDC dimana metode untuk *final cluster* menggunakan metode ROCK dengan langkah-langkah berikut ini,

A. Membagi (*splitting*) Original Data

Membagi data adalah memisahkan variabel-variabel yang digunakan menjadi sub-data yang keseluruhan berskala numerik dan keseluruhan berskala kategorik.

B. Pengelompokan Variabel Numerik Menggunakan Metode Hirarki *Agglomerative*

- a. Melakukan inisialisasi objek sebagai kelompok dengan anggota tunggal,
- b. Menentukan ukuran kemiripan dengan jarak *Euclidean* dengan rumus pada persamaan 2.3 dan membuat matriks jarak berukuran $n \times n$,

- c. Menggabungkan kelompok yang memiliki jarak terdekat,
- d. Memperbarui matriks jarak dengan metode *single linkage* seperti pada persamaan 2.4,
- e. Mengulangi langkah (c) dan (d) sampai hanya terbentuk 1 (satu) kelompok,
- f. Menghitung indeks validitas kelompok (*R-Square* dan *F*),
- g. Menentukan kandidat jumlah kelompok yang optimum berdasarkan indeks validitas yang diperoleh pada langkah (f),
- h. Melakukan kembali proses pengelompokan dengan *cutting-off* jumlah kelompok optimum yang diperoleh dari langkah (g),
- i. Mengulangi langkah (a) sampai dengan (h) menggunakan metode *complete linkage* seperti pada persamaan 2.5,
- j. Mengulangi langkah (a) sampai dengan (h) menggunakan metode *average linkage* seperti pada persamaan 2.6,
- k. Menghitung rasio S_W dan S_B dengan rumusan pada persamaan 2.15 dan 2.16, untuk *single linkage*, *complete linkage* dan *average linkage*,
- l. Membandingkan hasil langkah (k) dan menentukan kelompok terbaik untuk ukuran jarak *Euclidean* dengan kriteria rasio S_W dan S_B terkecil.

C. Pengelompokan Variabel Kategorik Menggunakan Metode ROCK

- a. Melakukan inisialisasi objek sebagai kelompok dengan anggota tunggal,
- b. Membentuk *similarity* antara objek dengan kriteria menggunakan persamaan 2.11,
- c. Menentukan *threshold* (θ) dalam matriks *adjacency*,
- d. Menghitung nilai *link* antara pengamatan,
- e. Menghitung *local heap* yaitu nilai *goodness measure* menggunakan persamaan 2.12,
- f. Menentukan *global heap* yaitu nilai maksimum *goodness measure* antara kolom di baris ke-i,
- g. Ulangi langkah (e) dan (f), sehingga diperoleh nilai maksimum di *global heap* dan *local heap*,

- h. Selama ukuran data $> k$, dengan k adalah jumlah kelompok yang ditentukan maka lakukan penggabungan kelompok yang memiliki nilai *local heap* terbesar dengan *global heap* terbesar menjadi satu kelompok, kemudian tambahkan *link* antar kelompok yang digabungkan, selanjutnya hapus kelompok yang digabungkan dari *local heap* dan perbaruan nilai *global heap* dengan hasil penggabungan,
- i. Lakukan langkah (h), sehingga diperoleh jumlah kelompok yang diharapkan atau tidak terdapat *link* antar kelompok,
- j. Mengulangi langkah (a) sampai dengan (i) dengan nilai θ berbeda,
- k. Menghitung rasio S_W dan S_B untuk masing-masing nilai θ dengan rumus seperti pada persamaan 2.23 dan 2.24,
- l. Membandingkan hasil langkah (k) untuk masing-masing nilai θ dan menentukan jumlah kelompok yang optimum dengan kriteria rasio S_W dan S_B terkecil.

D. Penggabungan Hasil Pengelompokan (Tahapan Ensemble)

Setelah mendapatkan kelompok yang optimum hasil metode ROCK dan *Agglomerative*, tahapan selanjutnya adalah melakukan penggabungan kelompok. Tahapan ini sama dengan melakukan pengelompokan data kategorik menggunakan metode ROCK, dimana *input* untuk tahapan ini adalah kelompok hasil metode hirarki *agglomerative* (*output* 1) dan kelompok hasil metode ROCK (*output* 2). *Output* 1 dan *output* 2 dinyatakan sebagai variabel kategorik yang digunakan untuk menyusun *final cluster* menggunakan metode ROCK. Jumlah *final cluster* optimum adalah jumlah kelompok yang memiliki rasio S_W dan S_B terkecil. Nilai ratio dihitung dengan rumusan seperti pada persamaan 2.23 dan 2.24.

E. Membentuk Rancangan Program Ensemble ROCK pada Software R

Perancangan program pada *software R* dilakukan dengan membentuk fungsi analisis dengan menggunakan bantuan beberapa fungsi-fungsi analisis yang terdapat dalam *package* analisis yang telah tersedia.

3.1.2 Menyusun Algoritma dan Program Metode Ensembel SWFM

Metode ensembel yang digunakan adalah algoritma CEBMDC dimana metode untuk *final cluster* berdasarkan *similarity weight* dan *filter method* dengan langkah-langkah berikut ini,

A. Membagi (*splitting*) Original Data

Membagi data adalah memisahkan variabel-variabel yang digunakan menjadi sub-data yang keseluruhan berskala numerik dan keseluruhan berskala kategorik.

B. Pengelompokan Variabel Numerik Menggunakan Metode Hirarki *Agglomerative*

Untuk pengelompokan berdasarkan variabel numerik ini dilakukan dengan tahapan yang sama dengan pengelompokan ensembel berdasarkan ROCK, seperti pada tahapan 2 subbab 3.1.1.

C. Pengelompokan Variabel Kategorik Menggunakan Metode ROCK

Untuk pengelompokan berdasarkan variabel kategorik ini dilakukan dengan tahapan yang sama dengan pengelompokan ensembel berdasarkan ROCK, seperti pada tahapan 3 subbab 3.1.1.

D. Penggabungan Hasil Pengelompokan (Tahapan Ensembel)

Setelah mendapatkan kelompok yang optimum hasil metode ROCK dan *Agglomerative*, tahapan selanjutnya adalah melakukan penggabungan kelompok. Tahapan ini melakukan pengelompokan dengan jarak yang digunakan adalah jarak pada persamaan 2.13, dimana *input* untuk tahapan ini adalah kelompok hasil metode hirarki *agglomerative* (*output* 1) dan kelompok hasil metode ROCK (*output* 2). *Output* 1 dan *output* 2 dinyatakan sebagai variabel kategorik yang digunakan untuk menyusun *final cluster* menggunakan ukuran jarak pada persamaan 2.13 serta algoritma *filter* pada persamaan 2.14. Jumlah *final cluster* optimum adalah jumlah kelompok yang memiliki rasio S_W dan S_B terkecil. Nilai ratio dihitung dengan rumusan seperti pada persamaan 2.23 dan 2.24.

E. Membentuk Rancangan Program Ensemble SWFM pada *Software R*

Perancangan program pada *software R* dilakukan dengan membentuk fungsi analisis dengan menggunakan bantuan beberapa fungsi-fungsi analisis yang terdapat dalam *package* analisis yang telah tersedia.

3.2 Kajian Empiris

Kajian empiris merupakan kajian yang berhubungan dengan studi kasus. Kajian teoritis dilakukan untuk mencapai tujuan ketiga dan keempat dari tujuan penelitian. Kajian empiris yang dilakukan meliputi data penelitian, variabel penelitian yang digunakan, serta tahapan analisis data. Data dan variabel penelitian ditentukan berdasarkan dengan studi kasus yang telah dipilih yaitu pengelompokan aksesori jeruk hasil fusi protoplasma.

3.2.1 Data Penelitian

Data yang digunakan dalam penelitian ini adalah data sekunder yang dikutip dari data penelitian Irfandi (2016). Data tersebut merupakan data hasil pengamatan yang dilakukan oleh Balitjestro Kota Batu, Malang. Pengamatan dilakukan pada aksesori jeruk hasil fusi protoplasma dari induk *Satsuma Mandarin* dan *Siam Madu*. Data yang digunakan merupakan 25 aksesori yang dibedakan antara data numerik dan data kategorik

3.2.2 Identifikasi Variabel Penelitian

Variabel penelitian dalam penelitian ini menggunakan data karakteristik tanaman jeruk yang disesuaikan dengan aturan pengamatan buah jeruk yang ada di Balitjestro. Variabel yang menunjukkan data numerik buah jeruk ditampilkan pada Tabel 3.1, dan variabel yang menunjukkan data kategorik buah jeruk ditampilkan pada Tabel 3.2. Pengambilan data setiap variabel yang dilakukan oleh Balitjestro adalah dengan melakukan pengamatan terhadap 10 buah dari setiap aksesori atau pohon jeruk. Setiap buah yang diamati kemudian dilihat karakteristiknya berdasarkan seluruh variabel penelitian yang telah ditentukan. Setelah diperoleh hasil pengamatan terhadap masing-masing buah, selanjutnya dihitung nilai pengamatan untuk setiap aksesori. Untuk data numerik, nilai yang digunakan adalah

nilai rata-rata sehingga nilai untuk setiap aksesori merupakan nilai rata-rata dari 10 sampel buah, sedangkan untuk data kategorik, nilai yang digunakan adalah nilai modus sehingga nilai untuk setiap aksesori merupakan modus dari 10 sampel buah.

Tabel 3.1 Variabel data numerik buah jeruk

Variabel		Satuan
Diameter Buah	(X ₁)	milimeter
Tebal Kulit	(X ₂)	milimeter
Jumlah Juring	(X ₃)	buah
Tebal Daging Kulit	(X ₄)	milimeter
Diameter <i>Axis</i>	(X ₅)	milimeter
Jumlah Biji	(X ₆)	buah
Berat Buah	(X ₇)	gram
<i>Brix</i>	(X ₈)	derajat

Tabel 3.2 Variabel data kategorik buah jeruk

No	Variabel	Skala	Keterangan
1	Bentuk buah (X ₉)	1	Spheroid
		2	Elipsoid
		3	Pyriform
		4	Oblique
		5	Obloid
		6	Ovoid
2	Warna Kulit (X ₁₀)	1	Orange
		2	Orange Kekuningan
		3	Kuning Keorangean
		4	Kuning Kehijauan
		5	Hijau Kekuningan
		6	Hijaumuda
3	Permukaan Kulit (X ₁₁)	1	Halus
		2	Berpori
		3	Berambut
		4	Kasar
		5	Bergelombang
4	<i>Axis</i> Buah (X ₁₂)	1	Padat
		2	Agak Berlubang
		3	Berlubang

Tabel 3.2 Variabel data kategorik buah jeruk (Lanjutan)

No	Variabel	Skala	Keterangan
5	Warna Daging Kulit (X_{13})	1	Putih
		2	Kuning
		3	Lainnya
6	Tampak <i>Axis</i> (X_{14})	1	Bulat
		2	Tidak Beraturan
7	Tekstur <i>Pulp</i> (X_{15})	1	Lembut
		2	Sedang
		3	Keras

3.2.3 Tahapan Analisis Data

Analisis untuk studi kasus yang dilakukan untuk mencapai tujuan ketiga dan keempat penelitian pada penelitian ini dikelompokkan menjadi dua langkah analisis utama. Setiap langkah analisis ditujukan untuk menjawab masing-masing tujuan penelitian. Tahapan analisis data yang dilakukan dalam bentuk diagram alir disajikan pada Gambar 3.1

3.2.3.1 Mengelompokkan Aksesori Jeruk Hasil Fusi Protoplasma dengan Metode Ensemble ROCK dan Ensemble SWFM

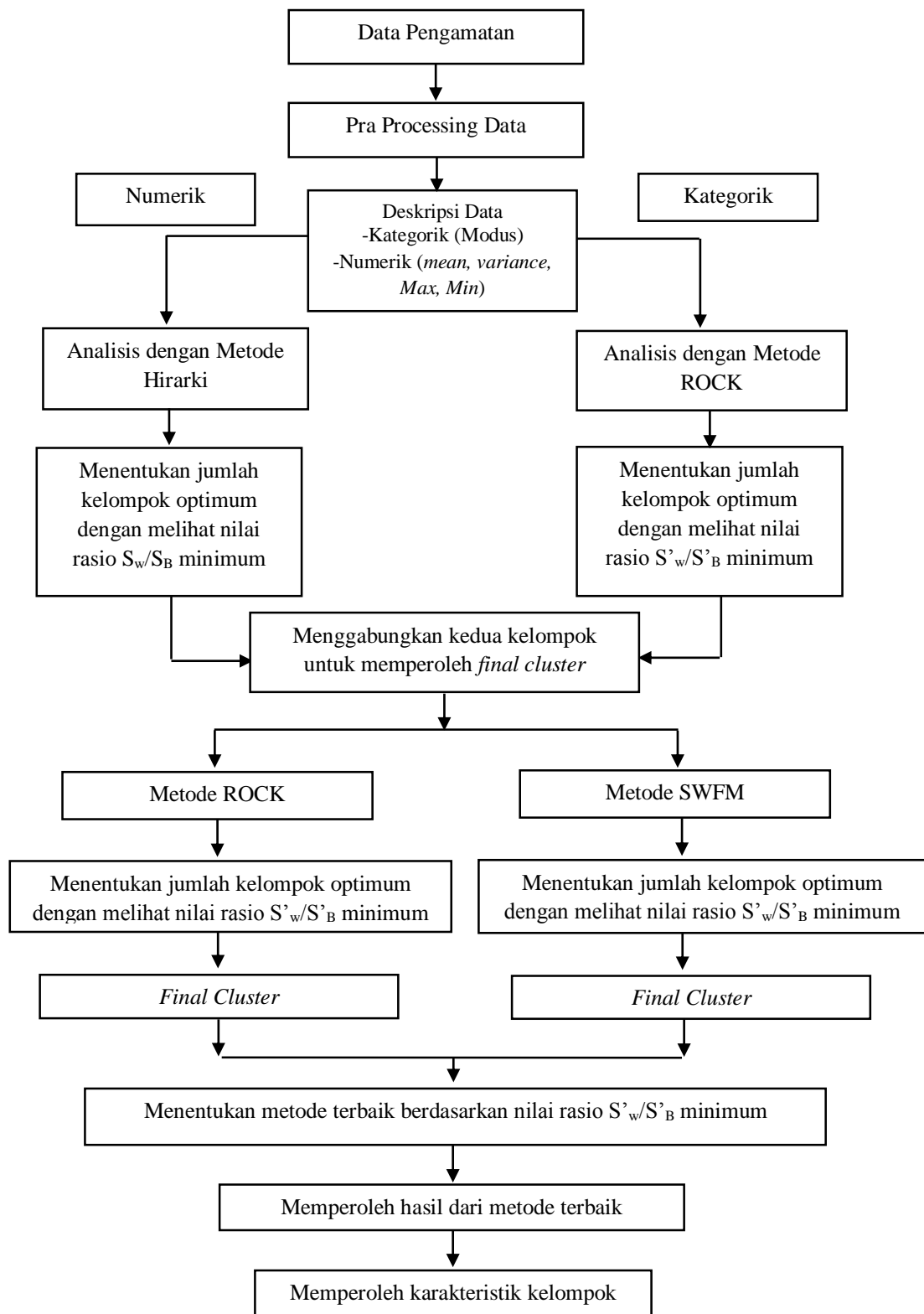
Berikut ini merupakan tahapan implementasi program pengelompokan ensemble untuk mengelompokkan aksesori jeruk hasil fusi protoplasma antara lain,

- Membagi data dengan 15 variabel menjadi 8 variabel berskala numerik (X_1 - X_8) dan 7 variabel berskala kategorik (X_9 - X_{15}),
- Untuk data dengan skala numerik dilakukan pengelompokan dengan metode hirarki *agglomerative*. Jarak yang digunakan adalah jarak *Euclidean* dan pautan yang digunakan adalah *single linkage*, *complete linkage* dan *average linkage*, seperti tahapan 2 pada subbab 3.1.1.,
- Melakukan perhitungan rasio S_W dan S_B untuk menentukan pautan yang digunakan dengan jumlah kelompok optimum untuk metode hirarki *agglomerative* yang dinyatakan sebagai *output* 1, seperti tahapan 2 bagian (k) dan (l) pada subbab 3.1.1.,

- d. Untuk data dengan skala kategorik dilakukan pengelompokan dengan metode ROCK dengan nilai θ sebesar 0,05; 0,07; 0,10; 0,12; 0,15; 0,17; 0,20; 0,22; 0,25; 0,27; dan 0,30, seperti tahapan 3 pada subbab 3.1.1.,
- e. Melakukan perhitungan rasio S_W dan S_B untuk menentukan nilai θ optimum untuk metode metode ROCK yang dinyatakan sebagai *output 2*, seperti tahapan 3 bagian (k) dan (l) pada subbab 3.1.1.,
- f. Untuk metode ensemble berdasarkan ROCK, dilakukan pengabungan *output 1* dan *output 2* yang kemudian dikelompokkan menggunakan metode ROCK untuk menghasilkan *final cluster*, seperti tahapan 4 pada subbab 3.1.1.,
- g. Untuk metode ensemble berdasarkan SWFM, dilakukan pengabungan *output 1* dan *output 2* yang kemudian dikelompokkan menggunakan metode *similarity weight and filter method* untuk menghasilkan *final cluster*, seperti tahapan 4 pada subbab 3.1.2.

3.2.3.2 Melakukan Perbandingan Antara Metode Ensemble ROCK dengan Ensemble SWFM

Pada tahapan ini dilakukan perbandingan dari hasil yang dipeoleh dari subbab 3.2.3.1 bagian (f) dengan hasil dari subbab 3.2.3.1 bagian (g). Perbandingan dilakukan untuk melihat kinerja dari kedua metode dalam menganalisis data berskala campuran numerik dan kategorik. Metode yang memberikan kinerja lebih baik dapat dilihat dari nilai rasio S_W dan S_B , dimana metode terbaik adalah metode dengan nilai rasio terkecil. Setelah diperoleh metode terbaik, selanjutnya dideskripsikan karakteristik dari setiap kelompok yang dibentuk berdasarkan variabel-variabel penelitiannya. Untuk mengetahui apakah kelompok yang dihasilkan berbeda atau tidak, maka dilakukan pengujian perbedaan rata-rata antar kelompok dengan metode MANOVA.



Gambar 3.1 Diagram alir analisis data menggunakan metode ensemble ROCK dan ensemble SWFM.

“Halaman ini sengaja dikosongkan”

BAB IV

ANALISIS DAN PEMBAHASAN

Pada bab ini dijelaskan mengenai analisis dan pembahasan untuk menjawab tujuan-tujuan pada penelitian ini. Pada bagian awal bab ini membahas algoritma dan program yang digunakan untuk metode ensemble ROCK dan ensemble SWFM. Analisis dilakukan dengan menggunakan bantuan *software R Project*. Bagian berikutnya dipaparkan hasil implementasi program dalam menyelesaikan studi kasus berupa pengelompokan aksesori jeruk hasil fusi protoplasma. Bagian akhir pada bab ini membahas perbandingan kinerja hasil dari ensemble ROCK dan ensemble SWFM.

4.1 Penyusunan Algoritma dan Program Metode Ensemble ROCK

Pada subbab ini dijelaskan mengenai algoritma dan program aplikasi yang dilakukan untuk menjawab tujuan pertama dalam penelitian. Penyusunan algoritma dilakukan berdasarkan metode penelitian yang telah dibahas dalam Bab III subbab 3.2.1, algoritma tersebut kemudian dijadikan dasar dalam penyusunan program aplikasi pada *software R Project*.

4.1.1 Algoritma Pengelompokan Ensemble ROCK

Algoritma pada analisis pengelompokan ensemble ROCK dapat dijelaskan sebagai berikut, pertama dilakukan pembagian data (*splitting*) menjadi data murni numerik dan data murni kategorik. Misalkan terdapat data dengan variabel berskala campuran sebanyak m , dengan $m_{numerik}$ merupakan jumlah variabel numerik, dan $m_{kategorik}$ merupakan jumlah variabel kategorik, sehingga diperoleh $m = m_{numerik} + m_{kategorik}$. Setelah terbentuk dua subdata, kemudian dilakukan pengelompokan secara terpisah dimana data numerik dikelompokkan menggunakan metode hirarki *agglomerative* dan data kategorik dikelompokkan menggunakan metode ROCK. Tahap berikutnya adalah hasil kedua pengelompokan tersebut digabungkan sebagai

data kategorik yang kemudian dikelompokkan menggunakan metode ensemble berdasarkan ROCK untuk diperoleh kelompok akhir (*final cluster*). Prosedur untuk analisis ini dapat dijabarkan dalam bentuk *flowchart* seperti ditunjukkan pada Gambar 4.1. *Input* yang diberikan dalam analisis pengelompokan ensemble ROCK merupakan matriks **D** sebagai berikut,

$$\mathbf{D} = \left[\begin{array}{cccc|cccc} x_{11} & x_{12} & \cdots & x_{1m_{\text{numerik}}} & x_{11} & x_{12} & \cdots & x_{1m_{\text{kategorik}}} \\ x_{21} & x_{22} & \cdots & x_{2m_{\text{numerik}}} & x_{21} & x_{22} & \cdots & x_{2m_{\text{kategorik}}} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nm_{\text{numerik}}} & x_{n1} & x_{n2} & \cdots & x_{nm_{\text{kategorik}}} \end{array} \right]$$

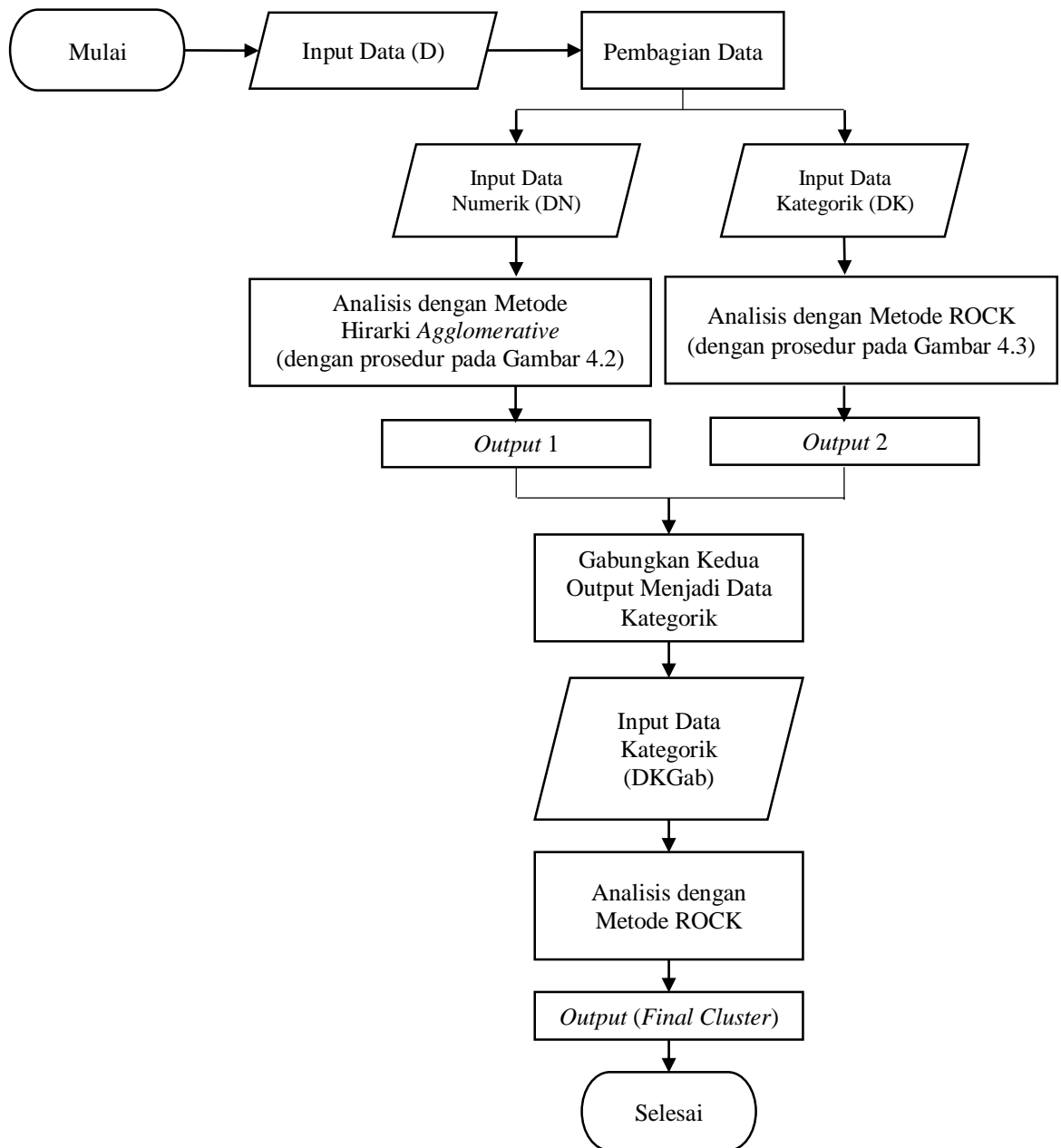
Output yang diperoleh merupakan hasil pengelompokan yang optimum dimana setiap objek pengamatan masuk dalam kelompok tertentu dengan nilai ratio S_W dan S_B terkecil. Dalam analisis pengelompokan ensemble ROCK dapat dilakukan simulasi dengan melakukan pengelompokan berdasarkan beberapa nilai θ yang berbeda.

Prosedur dalam analisis ensemble ROCK ini merupakan prosedur yang didalamnya tersarang prosedur lain, yaitu pengelompokan data numerik dan pengelompokan data kategorik. Prosedur pengelompokan untuk data numerik dibentuk berdasarkan tahapan analisis pada Bab III subbab 3.2.1 tahapan kedua. Prosedur pengelompokan data numerik yang dilakukan dalam penelitian ini dalam bentuk *flowchart* seperti ditunjukkan pada Gambar 4.2. Prosedur analisis metode hirarki *agglomerative* memerlukan *input* berupa data numerik berukuran n objek dengan m_{numerik} variabel. *Input* yang diberikan dalam analisis pengelompokan analisis metode hirarki *agglomerative* merupakan matriks **DN** sebagai berikut,

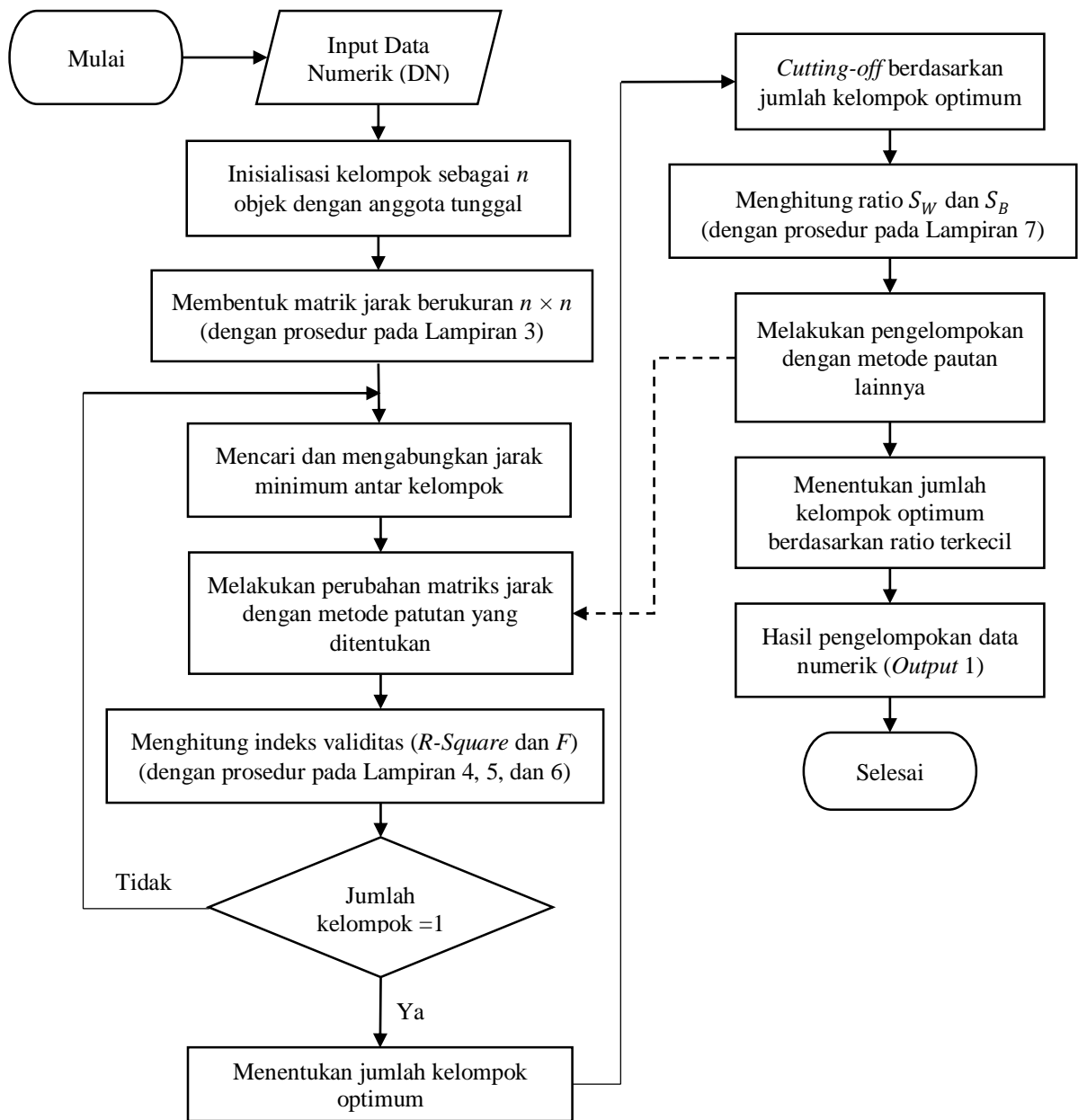
$$\mathbf{DN} = \left[\begin{array}{cccc} x_{11} & x_{12} & \cdots & x_{1m_{\text{numerik}}} \\ x_{21} & x_{22} & \cdots & x_{2m_{\text{numerik}}} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nm_{\text{numerik}}} \end{array} \right]$$

Output yang diperoleh merupakan hasil pengelompokan yang optimum dimana setiap objek pengamatan masuk dalam kelompok tertentu dengan nilai ratio S_W dan

S_B terkecil. Dalam pengelompokan metode hirarki *agglomerative* dapat dilakukan simulasi dengan melakukan pengelompokan berdasarkan beberapa teknik pengelompokan (pautan) yang berbeda dan beberapa jumlah kelompok yang berbeda.



Gambar 4.1 *Flowchart* prosedur pengelompokan pada metode ensemble ROCK.



Gambar 4.2 *Flowchart* prosedur pengelompokan pada metode hirarki *agglomerative*.

Dalam prosedur analisis pengelompokan data numerik (Gambar 4.2) tersarang prosedur analisis lain yaitu,

- Perhitungan jarak antar objek dengan formula jarak *euclidean* dengan rumusan seperti pada persamaan 2.3.

$$d_{ij} = \sqrt{(\mathbf{x}_i - \mathbf{x}_j)^T (\mathbf{x}_i - \mathbf{x}_j)}$$

Persamaan diatas digunakan untuk menghitung jarak dua observasi dengan variabel-variabel berdimensi $m_{numerik}$ yaitu dengan obeservasi pertama berupa $\mathbf{x}_i = [x_1, x_2, \dots, x_{m_{numerik}}]^T$ dan obeservasi kedua berupa $\mathbf{x}_j = [x_1, x_2, \dots, x_{m_{numerik}}]^T$. Perhitungan jarak tersebut memiliki *flowchart* seperti ditunjukkan pada Lampiran 3.

- b. Perhitungan SST, SSW, *R-Square*, dan *F* dengan rumusan seperti berikut,

$$SST = \sum_{l=1}^{m_{numerik}} \sum_{i=1}^n (x_{il} - \bar{x}_l)^2$$

$$SSW = \sum_{c=1}^C \sum_{l=1}^{m_{numerik}} \sum_{i=1}^{n_c} (x_{ilc} - \bar{x}_{lc})^2$$

$$R-Square = \frac{[SST - SSW]}{SST}$$

$$F = \frac{[SST - SSW]}{SSW}$$

SST merupakan jumlah kuadrat selisih antara nilai individu dengan rata-rata total, sedangkan SSW merupakan variansi yang ada dalam kelompok masing-masing. Dengan C merupakan jumlah kelompok yang dibentuk dalam pengamatan, n merupakan total jumlah objek pengamatan, n_c merupakan jumlah anggota pada kelompok ke- c untuk $c = 1, 2, \dots, C$, \bar{x}_l merupakan rata-rata keseluruhan objek pada variabel ke- l untuk $l = 1, 2, \dots, m_{numerik}$, dan \bar{x}_{lc} merupakan rata-rata variabel ke- l pada kelompok ke- c . Perhitungan indeks validitas tersebut memiliki *flowchart* seperti ditunjukan pada Lampiran 4, Lampiran 5, dan Lampiran 6.

- c. Perhitungan rasio S_W dan S_B dengan rumusan seperti pada persamaan 2.15 dan 2.16.

$$S_W = \frac{1}{C} \sum_{c=1}^C S_c$$

$$S_B = \left[\frac{1}{C-1} \sum_{c=1}^C (\bar{x}_c - \bar{x})^2 \right]^{1/2}$$

Dengan S_c merupakan simpangan baku kelompok ke- c , \bar{x}_c adalah rata-rata kelompok ke- c , \bar{x} adalah rata-rata keseluruhan kelompok. dan C adalah jumlah kelompok yang terbentuk. Perhitungan ratio tersebut memiliki *flowchart* seperti ditunjukkan pada Lampiran 7.

Prosedur pengelompokan untuk data kategorik dibentuk berdasarkan tahapan analisis pada Bab III subbab 3.2.1 tahapan ketiga. Prosedur analisis untuk pengelompokan data kategorik yang digunakan adalah prosedur metode ROCK. Prosedur pengelompokan data kategorik yang dilakukan dalam penelitian ini dalam bentuk *flowchart* seperti ditunjukkan pada Gambar 4.3. Prosedur ini memerlukan *input* berupa data kategorik berukuran n objek dengan $m_{kategorik}$ variabel, nilai *threshold* (θ) dan jumlah kelompok yang diharapkan (k). *Input* yang diberikan dalam analisis pengelompokan ROCK merupakan matriks **DK** sebagai berikut,

$$\mathbf{DK} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1m_{kategorik}} \\ x_{21} & x_{22} & \cdots & x_{2m_{kategorik}} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nm_{kategorik}} \end{bmatrix}$$

Output yang diperoleh merupakan hasil pengelompokan yang optimum dimana setiap objek pengamatan masuk dalam kelompok tertentu dengan nilai ratio S_W dan S_B terkecil. Dalam analisis pengelompokan ROCK, dapat dilakukan simulasi dengan melakukan pengelompokan berdasarkan beberapa nilai θ yang berbeda.

Dalam prosedur analisis pengelompokan data kategorik (Gambar 4.3) tersarang prosedur lain yaitu,

- a. Perhitungan jarak antar objek (*similarity*) seperti pada persamaan 2.11.

$$sim(X_i, X_j) = \frac{|X_i \cap X_j|}{|X_i \cup X_j|}, \quad i \neq j$$

Persamaan diatas digunakan untuk menghitung jarak dua observasi dengan variabel - variabel berdimensi $m_{kategorik}$ yaitu dengan obeservasi pertama berupa himpunan $X_i = \{x_{1i}, x_{2i}, x_{3i}, \dots, x_{m_{kategorik}i}\}$ dan obeservasi kedua berupa himpunan $X_j = \{x_{1j}, x_{2j}, x_{3j}, \dots, x_{m_{kategorik}j}\}$. Perhitungan jarak tersebut memiliki *flowchart* seperti ditunjukkan pada Lampiran 8.

- b. Perhitungan *link* antar objek dengan *flowchart* seperti ditunjukkan pada Lampiran 9. Cara untuk menghitung *link* untuk semua kemungkinan pasangan dari n objek dapat menggunakan matriks \mathbf{A} . Matriks tetangga \mathbf{A} merupakan matriks berukuran $n \times n$ dengan $\mathbf{A}[X_i, X_j]$ bernilai 1 jika X_i dan X_j dinyatakan mirip (tetangga) dan bernilai 0 dan jika X_i dan X_j tidak mirip (bukan tetangga) dimana X_i dan X_j merupakan tetangga jika nilai $sim(X_i, X_j) \geq \theta$. Jumlah *link* antar pasangan X_i dan X_j diperoleh dari

$link(X_i, X_j) = \sum_{i=1}^n \mathbf{A}[X_i, :] \times \mathbf{A}[:, X_j]$. Dengan menggunakan nilai *link* yang diperoleh dapat dibentuk matrik *link* \mathbf{L} yang merupakan perkalian matriks tetangga \mathbf{A} dengan dirinya sendiri.

- c. Perhitungan *goodness measure* antar objek seperti pada persamaan 2.12, dengan *flowchart* seperti ditunjukkan pada Lampiran 10.

$$g(C_i, C_j) = \frac{link[C_i, C_j]}{(n_i + n_j)^{1+2f(\theta)} - n_i^{1+2f(\theta)} - n_j^{1+2f(\theta)}}$$

Dengan $link[C_i, C_j] = \sum_{X_i \in C_i, X_j \in C_j} link(X_i, X_j)$ yang menyatakan jumlah *link*

dari semua kemungkinan pasangan objek yang ada dalam C_i dan C_j , serta n_i dan n_j masing-masing menyatakan jumlah anggota dalam kelompok ke i dan

j , sedangkan $f(\theta) = \frac{1-\theta}{1+\theta}$.

- d. Perhitungan rasio S_W dan S_B dengan *flowchart* seperti ditunjukkan pada Lampiran 11. Rumusan nilai S_W yang digunakan adalah rumusan pada persamaan 2.23, sedangkan rumusan nilai S_B yang digunakan adalah rumusan pada persamaan 2.24.

$$S_W = [MSW]^{1/2}$$

$$S_B = [MSB]^{1/2}$$

Dengan memasukkan rumusan MSW pada persamaan 2.21 dan rumusan SSW pada persamaan 2.18, maka rumusan nilai S_W yang diperoleh adalah

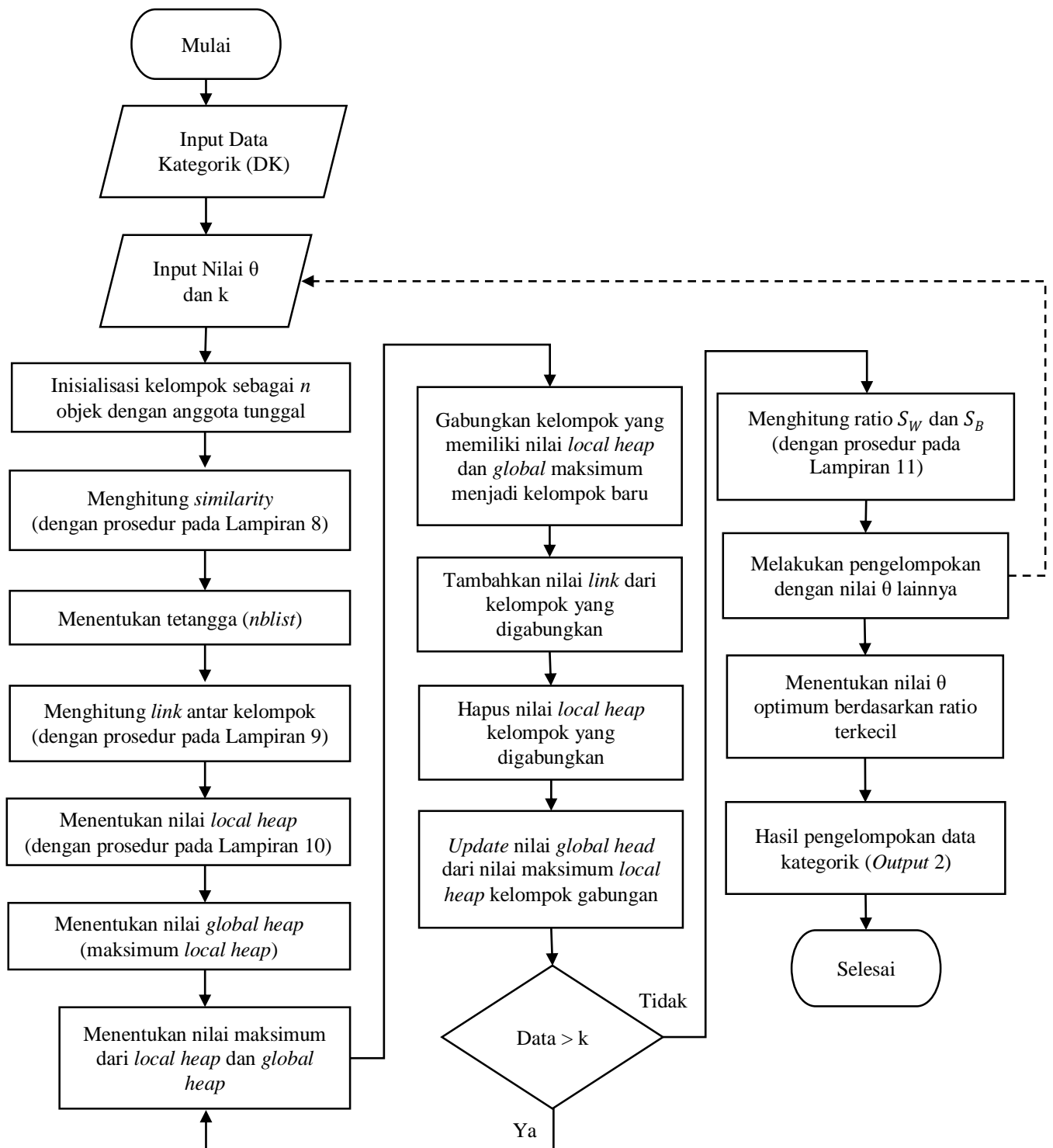
$$S_W = \left[\frac{\frac{n}{2} - \frac{1}{2} \sum_{c=1}^C \frac{1}{n_{.c}} \sum_{k=1}^K n_{kc}^2}{(n - c)} \right]^{1/2}$$

Dengan memasukkan rumusan MSB pada persamaan 2.22 dan rumusan SSB pada persamaan 2.19, maka rumusan nilai S_B yang diperoleh adalah

$$S_B = \left[\frac{\frac{1}{2} \left(\sum_{c=1}^C \frac{1}{n_{.c}} \sum_{k=1}^K n_{kc}^2 \right) - \frac{1}{2n} \sum_{k=1}^K n_{k.}^2}{(c - 1)} \right]^{1/2}$$

Dengan n merupakan jumlah pengamatan dengan n_k merupakan jumlah pengamatan dengan kategori ke- k dimana $k = 1, 2, 3, \dots, K$ dan $\sum_{k=1}^K n_k = n$.

Selanjutnya, n_{kc} merupakan jumlah pengamatan dengan kategori ke- k dan kelompok ke- c , dimana $c = 1, 2, 3, \dots, C$, sehingga $n_{.c} = \sum_{k=1}^K n_{kc}$ merupakan jumlah pengamatan pada kelompok ke- c dan $n_{k.} = \sum_{c=1}^C n_{kc}$ merupakan jumlah pengamatan pada kategori ke- k .



Gambar 4.3 Flowchart prosedur pengelompokan pada metode ROCK.

4.1.2 Algoritma Pemograman Ensemble ROCK di *R Project*

Untuk mempermudah dilakukannya pengelompokan, dibangun rumusan algoritma dalam bahasa pemograman (*syntax*) yang mengakomodir pengelompokan ensemble dengan pendekatan metode ROCK untuk pengelompokan data campuran. Dengan menggunakan program statistika *R-Project* (paket R) versi 3.3.1, pemograman untuk pengelompokan ensemble ROCK menggunakan *syntax* yang dilampirkan pada Lampiran 19. Pemogram ensemble ROCK memiliki langkah-langkah utama sebagai berikut,

Input : Data D

Output : Hasil pengelompokan dimana setiap objek pengamatan masuk dalam kelompok tertentu, serta nilai ratio S_W dan S_B dari masing-masing kelompok.

Tahap 1 : Memisahkan data menjadi dua kelompok data yaitu Data Numerik (DN) dan Data Kategorik (DK).

Tahap 2 : Melakukan pengelompokan DN dengan menggunakan algoritma pengelompokan metode hirarki *agglomerative*.

Tahap 3 : Melakukan pengelompokan DK dengan menggunakan algoritma pengelompokan metode ROCK.

Tahap 4 : Mengabungkan hasil dari Tahap 2 dan Tahap 3 menjadi data kategorik (DKGab).

Tahap 5 : Melakukan pengelompokan DKGab dengan menggunakan algoritma pengelompokan metode ROCK.

Tahap 6 : Menentukan nilai θ terbaik dengan nilai ratio S_W dan S_B terkecil.

Pada Tahap 1, pemisahan dilakukan dengan cara mengabungkan data-data numerik (X_1 - X_8) menjadi satu kelompok data sendiri dan data-data kategorik (X_9 - X_{15}) menjadi satu kelompok data sendiri. Analisis dilakukan dengan menggunakan fungsi `data.frame()` untuk menggabungkan data.

Pada Tahap 2, pengelompokan data numerik dilakukan dengan jumlah kelompok yang dibentuk berada pada nilai dua sampai delapan kelompok yang dilakukan dengan *syntax* yang dilampirkan pada Lampiran 14. Analisis tersebut dilakukan dengan bantuan fungsi-fungsi berikut ini,

- a. `dist()`, untuk membentuk matriks jarak.
- b. `hclust()`, untuk melakukan pengelompokan metode hirarki.
- c. `cutree()`, untuk memotong *dendogram* pada k kelompok.

Selanjutnya dilakukan penentuan jumlah kelompok optimum berdasarkan *plot* nilai *R-Square* dilakukan dengan *syntax* yang dilampirkan pada Lampiran 15 dan *plot* nilai *F* dilakukan dengan *syntax* yang dilampirkan pada Lampiran 16. Selain penentuan jumlah kelompok dilakukan juga penentuan metode revisi jarak terbaik berdasarkan nilai ratio S_W dan S_B , pemilihan dilakukan dengan *syntax* yang dilampirkan pada Lampiran 17.

Pada Tahap 3, pengelompokan data kategorik dilakukan dengan *syntax* yang dilampirkan pada Lampiran 18. Analisis pengelompokan metode ROCK pada *software R* dilakukan dengan bantuan *package cba*, fungsi-fungsi yang digunakan adalah sebagai berikut,

- a. `as.dummy()`, untuk menyatakan data dalam bentuk *dummy*.
- b. `rockCluster()`, untuk melakukan pengelompokan.
- c. `fitted()`, untuk memunculkan hasil pengelompokan dalam bentuk prediksi kelompok.
- d. `aov()`, untuk memperoleh nilai SSW dan SSB dari analisis variansi.

Pengelompokan metode ROCK dengan *software R* dapat menghasilkan hasil pengelompokan yang berbeda setiap melakukan *running* data. Hal ini dikarenakan adanya perbedaan pengambilan nilai *goodness measure* ketika terdapat beberapa nilai *goodness measure* yang sama (diambil secara random). Oleh karena itu, dalam penelitian ini telah ditentukan bahwa hasil pengelompokan yang diambil sebagai bahan analisis merupakan hasil *running* pertama dari *syntax* yang digunakan.

Pada Tahap 4 sampai Tahap 6 dilakukan dengan *syntax* yang dilampirkan pada Lampiran 19 dengan *input* yang diberikan adalah data, jumlah kelompok numerik, jumlah kelompok kategorik dan nilai θ . Analisis pada tahap 5 menggunakan fungsi-fungsi yang telah digunakan pada tahap 3.

4.2 Penyusunan Algoritma dan Program Metode Ensemble SWFM

Pada subbab ini dijelaskan mengenai algoritma dan program aplikasi yang dilakukan untuk menjawab tujuan kedua dalam penelitian. Penyusunan algoritma dilakukan berdasarkan metode penelitian yang telah dibahas dalam Bab III subbab 3.2.2, algoritma tersebut kemudian dijadikan dasar dalam penyusunan program aplikasi pada *software R Project*.

4.2.1 Algoritma Pengelompokan Ensemble SWFM

Algoritma pada analisis pengelompokan ensemble SWFM memiliki tahapan yang sama dengan ensemble ROCK seperti yang telah dijelaskan pada Bab II. Tahapan pertama yang dilakukan adalah pembagian data (*splitting*) menjadi data murni numerik dan data murni kategorik. Misalkan terdapat data dengan variabel berskala campuran sebanyak m , dengan $m_{numerik}$ merupakan jumlah variabel numerik, dan $m_{kategorik}$ merupakan jumlah variabel kategorik, sehingga diperoleh $m = m_{numerik} + m_{kategorik}$. Setelah terbentuk dua subdata, kemudian dilakukan pengelompokan secara terpisah dimana data numerik dikelompokkan menggunakan metode hirarki *agglomerative* dan data kategorik dikelompokkan menggunakan metode ROCK. Tahap berikutnya adalah hasil kedua pengelompokan tersebut digabungkan sebagai data kategorik yang kemudian dikelompokkan menggunakan metode SWFM untuk diperoleh kelompok akhir (*final cluster*).

Prosedur dalam analisis ensemble SWFM ini merupakan prosedur yang didalamnya tersarang prosedur lain, yaitu pengelompokan data numerik dengan prosedur (Gambar 4.2) dan pengelompokan data kategorik (Gambar 4.3). Prosedur untuk analisis ini dapat dijabarkan dalam bentuk *flowchart* seperti ditunjukkan pada

Gambar 4.4. *Input* yang diberikan dalam analisis pengelompokan ensemble SWFM merupakan matriks **D** sebagai berikut,

$$\mathbf{D} = \left[\begin{array}{cccc|cccc} x_{11} & x_{12} & \cdots & x_{1m_{numerik}} & x_{11} & x_{12} & \cdots & x_{1m_{kategorik}} \\ x_{21} & x_{22} & \cdots & x_{2m_{numerik}} & x_{21} & x_{22} & \cdots & x_{2m_{kategorik}} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nm_{numerik}} & x_{n1} & x_{n2} & \cdots & x_{nm_{kategorik}} \end{array} \right]$$

Output yang diperoleh merupakan hasil pengelompokan yang optimum dimana setiap objek pengamatan masuk dalam kelompok tertentu dengan nilai ratio S_W dan S_B terkecil. Dalam analisis pengelompokan ensemble SWFM dapat dilakukan simulasi pengelompokan menjadi beberapa hasil dengan jumlah kelompok yang berbeda.

Selain prosedur pengelompokan data numerik dan prosedur pengelompokan data kategorik, terdapat prosedur lain yang tersarang dalam melakukan analisis pengelompokan ensemble SWFM. Prosedur tersebut antara lain,

- a. Perhitungan jarak antar objek (*similarity weight*) seperti pada persamaan 2.13.

$$sim(X_i, X_j) = \sum_{i \leq n_i, j \leq n_j} \frac{S_{ij}}{\max(n_i, n_j)}, \quad i \neq j,$$

$$\text{dengan } S_{ij} = \frac{|X_i \cap X_j|}{|X_i \cup X_j|}$$

Dimana $m_{kategorik}$ merupakan jumlah variabel kategorik dalam pengamatan, n merupakan total jumlah objek pengamatan, n_i merupakan jumlah anggota dalam kelompok ke- i , dan n_j merupakan jumlah anggota dalam kelompok ke- j . Persamaan diatas digunakan untuk menghitung jarak dua observasi dengan variabel - variabel berdimensi m yaitu dengan obeservasi pertama berupa himpunan $X_i = \{x_{1i}, x_{2i}, x_{3i}, \dots, x_{m_{kategorik}i}\}$ dan obeservasi kedua berupa himpunan $X_j = \{x_{1j}, x_{2j}, x_{3j}, \dots, x_{m_{kategorik}j}\}$. Perhitungan jarak tersebut memiliki *flowchart* seperti ditunjukkan pada Lampiran 12.

- b. Perhitungan nilai jarak F antar objek, seperti pada persamaan 2.14.

$$F(X_i, X_j) = \sum_{i=1}^{n_i} \sum_{j=1}^{n_j} w_{ij} d(X_i, X_j)$$

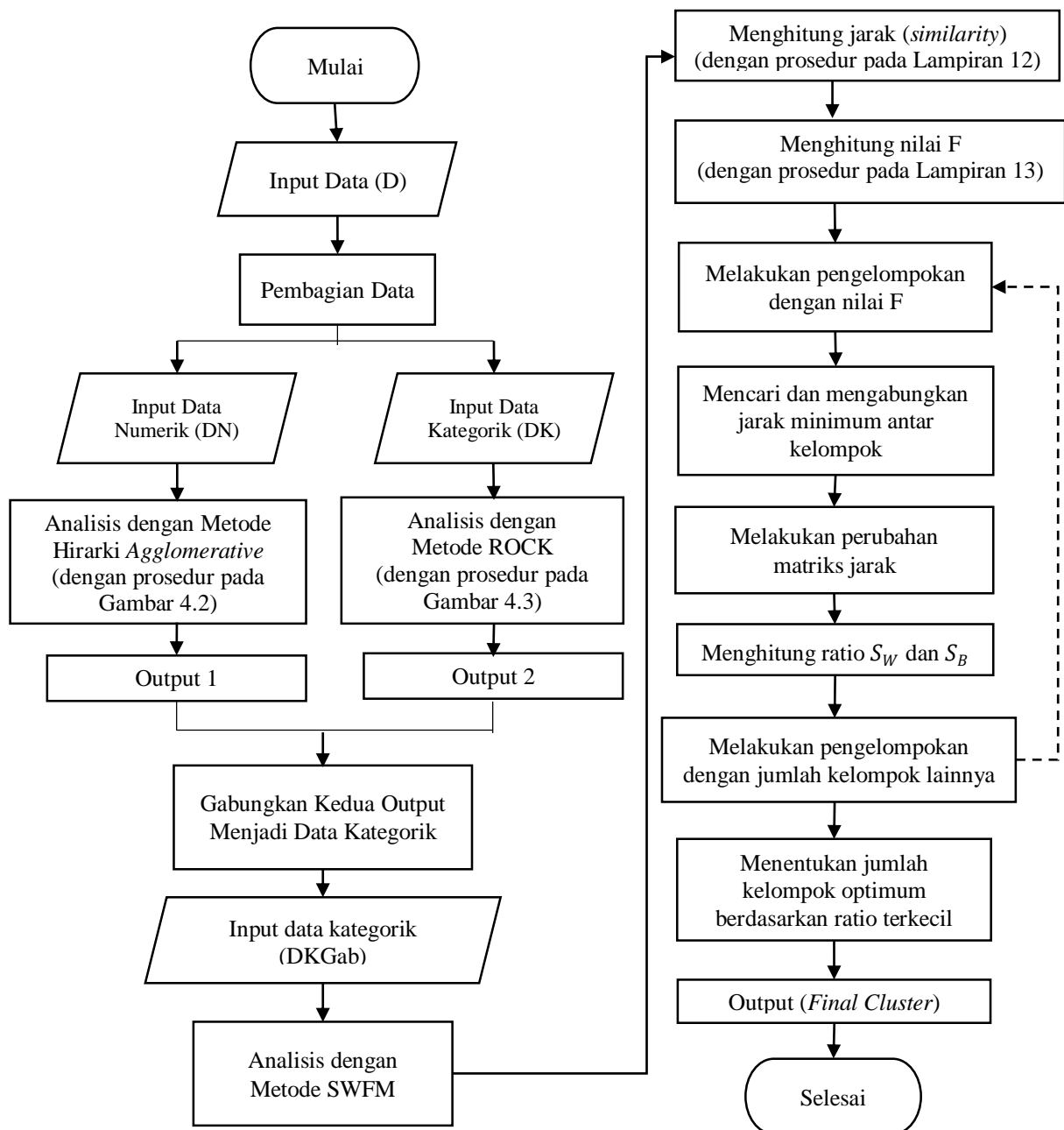
dengan $d(X_i, X_j) = 1 - \text{sim}(X_i, X_j)$ dan w_{ij} merupakan bobot antara kelompok ke i dan j yang bernilai 0,50. Persamaan diatas digunakan untuk menghitung ukuran tidakkemiripan dua observasi dengan variabel-variabel berdimensi m dengan jarak atau ukuran kemiripan yang diperoleh dari bagian (a). Perhitungan nilai jarak F tersebut memiliki *flowchart* seperti ditunjukkan pada Lampiran 13.

- c. Perhitungan rasio S_W dan S_B dengan *flowchart* seperti ditunjukkan pada Lampiran 11. Perhitungan nilai ratio ini menggunakan perhitungan yang sama dengan perhitungan ratio pada metode ROCK. Dengan rumusan S_W dan S_B yang digunakan adalah

$$S_W = \left[\frac{\frac{n}{2} - \frac{1}{2} \sum_{c=1}^C \frac{1}{n_{.c}} \sum_{k=1}^K n_{kc}^2}{(n - c)} \right]^{1/2}$$

$$S_B = \left[\frac{\frac{1}{2} \left(\sum_{c=1}^C \frac{1}{n_{.c}} \sum_{k=1}^K n_{kc}^2 \right) - \frac{1}{2n} \sum_{k=1}^K n_{k.}^2}{(c - 1)} \right]^{1/2}$$

Dengan n merupakan jumlah pengamatan dengan n_k merupakan jumlah pengamatan dengan kategori ke- k dimana $k = 1, 2, 3, \dots, K$ dan $\sum_{k=1}^K n_k = n$. Selanjutnya, n_{kc} merupakan jumlah pengamatan dengan kategori ke- k dan kelompok ke- c , dimana $c = 1, 2, 3, \dots, C$, sehingga $n_{.c} = \sum_{k=1}^K n_{kc}$ merupakan jumlah pengamatan pada kelompok ke- c dan $n_{k.} = \sum_{c=1}^C n_{kc}$ merupakan jumlah pengamatan pada kategori ke- k .



Gambar 4.4 Flowchart prosedur pengelompokan pada metode ensemble SWFM

4.2.2 Algoritma Pemograman Ensemble SWFM di R Project

Untuk mempermudah dilakukannya pengelompokan, dibangun rumusan algoritma dalam bahasa pemograman (*syntax*) yang mengakomodir pengelompokan ensemble dengan pendekatan metode SWFM untuk pengelompokan data campuran. Pemograman untuk analisis pengelompokan

ensemble SWFM memiliki kombinasi analisis yang lebih banyak dibandingkan dengan metode ensemble ROCK. Jika pada metode ensemble ROCK memiliki dua kombinasi analisis yaitu metode hirarki *agglomerative* dan metode ROCK, pada metode ensemble SWFM memiliki tiga kombinasi analisis yaitu metode hirarki *agglomerative*, metode ROCK, dan metode berdasarkan jarak *similarity weight* dengan *filter method*.

Dengan menggunakan program statistik *R-Project* (paket R) versi 3.3.1, pemograman untuk pengelompokan ensemble SWFM menggunakan *syntax* yang dilampirkan pada Lampiran 20. Pemogram ensemble SWFM memiliki langkah-langkah utama sebagai berikut,

Input : Data D
Output : Hasil pengelompokan dimana setiap objek pengamatan masuk dalam kelompok tertentu, serta nilai ratio S_W dan S_B dari masing-masing kelompok.

Tahap 1 : Memisahkan data menjadi dua kelompok data yaitu Data Numerik (DN) dan Data Kategorik (DK).
Tahap 2 : Melakukan pengelompokan DN dengan menggunakan algoritma pengelompokan metode hirarki *agglomerative*.
Tahap 3 : Melakukan pengelompokan DK dengan menggunakan algoritma pengelompokan metode ROCK.
Tahap 4 : Mengabungkan hasil dari Tahap 2 dan Tahap 3 menjadi data kategorik (DKGab).
Tahap 5 : Melakukan pengelompokan DKGab dengan menggunakan algoritma pengelompokan metode SWFM.
Tahap 6 : Menentukan jumlah kelompok terbaik dengan nilai ratio S_W dan S_B terkecil.

Pada Tahap 1 sampai Tahap 3, analisis dilakukan sama seperti yang dijelaskan pada subbab 4.1.2., sedangkan pada Tahap 4 sampai Tahap 6 dilakukan dengan *syntax* yang dilampirkan pada Lampiran 20. Untuk analisis pada Tahap 5,

analisis dilakukan tanpa menggunakan fungsi pembentukan matriks jarak yang ada (*dist*). Akan tetapi, pembentukan matriks jarak dianalisis dengan membentuk suatu fungsi matriks jarak yang disesuaikan dengan rumusan pada persamaan 2.13 dan 2.14.

4.3 Hasil dan Analisis Pengelompokan dengan Metode Ensemble ROCK dan Ensemble SWFM

Pada subbab ini dijelaskan mengenai hasil dan analisis yang dilakukan untuk menjawab tujuan ketiga dalam penelitian. Subbab ini difokuskan untuk menyelesaikan studi kasus mengenai pengelompokan aksesori jeruk hasil fusi protoplasma antara jeruk Siam Madu dan jeruk Satsuma Mandarin. Analisis dilakukan dengan menggunakan *syntax* pada *software R Project* yang telah dijelaskan pada subbab 4.1.2 dan 4.2.2.

4.3.1 Deskripsi Karakteristik Aksesori Jeruk Hasil Fusi Protoplasma

Hasil dari fusi protoplasma antara jeruk Siam Madu dan jeruk Satsuma Mandarin adalah 120 aksesori, akan tetapi hanya 25 aksesori yang berbuah matang sehingga dapat diamati. Pengamatan terhadap aksesori dilakukan dengan melihat karakteristik morfologi dari buah yang dihasilkan. Karakteristik tersebut terdiri dari 15 variabel dengan 8 variabel berskala numerik (data numerik) dan 7 variabel berskala kategorik (data kategorik). Berikut ini merupakan deskripsi karakteristik aksesori jeruk berdasarkan variabel pengamatannya,

4.3.1.1 Data Numerik

Data numerik pada karakteristik morfologi jeruk meliputi data-data dari variabel diameter buah, tebal kulit, jumlah juring, tebal daging kulit, diameter *axis*, jumlah biji, berat buah, dan nilai *brix*. Rata-rata diameter buah dari 25 objek pengamatan adalah sebesar 57,30 milimeter (mm) dengan diameter terkecil buah sebesar 47,16 mm dan diameter terbesar sebesar 69,91 mm (Tabel 4.1). Berdasarkan diameter buah tersebut (Gambar 4.5), diketahui bahwa aksesori yang dihasilkan dari proses fusi protoplasma memiliki ukuran buah yang beragam.

Rata-rata berat buah dari 25 objek pengamatan adalah sebesar 100,30 gram dengan berat terkecil buah sebesar 58,05 gram dan berat terbesar sebesar 165,99 gram (Tabel 4.1). Berdasarkan berat buah tersebut (Gambar 4.5), diketahui bahwa aksesori yang dihasilkan dari proses fusi protoplasma memiliki ukuran buah yang sangat beragam.

Tebal kulit buah merupakan salah satu variabel yang sangat mempengaruhi perubahan kandungan air dalam jeruk selama masa penyimpanan. Semakin tebal kulit jeruk menyebabkan semakin cepatnya penyerapan kandungan air dalam jeruk. Rata-rata tebal kulit buah dari 25 objek pengamatan adalah sebesar 1,83 mm dengan tebal kulit tertipis sebesar 1,26 mm dan tertebal sebesar 2,54 mm (Tabel 4.1). Berdasarkan tebal kulit tersebut (Gambar 4.5), diketahui bahwa aksesori yang dihasilkan dari proses fusi protoplasma memiliki ketebalan kulit buah yang beragam dengan satu data berupa *outlier*.

Seperti halnya tebal kulit, penyerapan kandungan air juga sangat dipengaruhi oleh tebal daging kulit. Rata-rata tebal daging kulit buah dari 25 objek pengamatan adalah sebesar 1,72 mm dengan tebal daging kulit tertipis sebesar 1,21 mm dan tertebal sebesar 2,62 mm (Tabel 4.1). Berdasarkan tebal daging kulit tersebut (Gambar 4.5), diketahui bahwa aksesori yang dihasilkan dari proses fusi protoplasma memiliki ketebalan daging kulit buah yang beragam dengan satu data berupa *outlier*.

Juring merupakan bagian buah jeruk yang berbentuk ruang atau berpetak-petak yang mudah dilepas atau dibuka dari bulatan buahnya. Rata-rata jumlah juring dari 25 objek pengamatan adalah sebanyak 10 buah dengan jumlah tersedikit yaitu 8 buah dan jumlah terbanyak yaitu 12 buah (Tabel 4.1). Berdasarkan jumlah juring yang dihasilkan (Gambar 4.5), diketahui bahwa aksesori yang dihasilkan dari proses fusi protoplasma memiliki jumlah juring yang beragam.

Biji jeruk merupakan salah satu bagian dari buah jeruk yang tidak disukai konsumen. Konsumen umumnya menyukai jenis buah jeruk dengan biji sedikit. Rata-rata jumlah biji dari 25 objek pengamatan adalah sebanyak 14 buah dengan jumlah tersedikit yaitu 4 buah dan jumlah terbanyak yaitu 22 buah (Tabel 4.1).

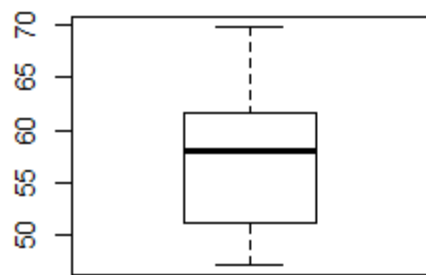
Berdasarkan jumlah biji yang dihasilkan (Gambar 4.5), diketahui bahwa aksesori yang dihasilkan dari proses fusi protoplasma memiliki jumlah biji yang sangat beragam.

Bagian *axis* dari buah jeruk merupakan bagian tengah diameter buah jeruk yang merupakan pusat dari juring jeruk. Rata-rata diameter *axis* buah dari 25 objek pengamatan adalah sebesar 12,84 mm dengan diameter terkecil buah sebesar 6,63 mm dan diameter terbesar sebesar 18,53 mm (Tabel 4.1). Berdasarkan diameter *axis* buah tersebut (Gambar 4.5), dapat diketahui bahwa aksesori yang dihasilkan dari proses fusi protoplasma memiliki ukuran *axis* buah yang beragam.

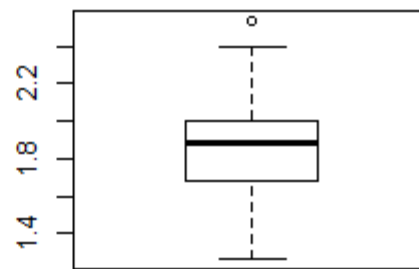
Brix atau derajat *brix* merupakan jumlah suatu zat padat semu yang larut (dalam gram) pada setiap 100 gram larutan. Milainya nilai *brix* jeruk 12 artinya dalam 100 gram larutan jeruk, 12 gram merupakan zat padat terlarut dan 88 gram sisanya adalah air. Zat padat terlarut pada buah jeruk merupakan fruktosa. Fruktosa yang biasa dikenal sebagai salah satu bentuk gula dalam buah-buahan dipercaya memiliki dampak negatif pada penderita diabetes. Selain itu, fruktosa berlebihan juga berdampak buruk bagi seseorang dengan berat badan yang berlebih. Rata-rata nilai *brix* dari 25 objek pengamatan adalah sebesar 11,80 dengan nilai terendah yaitu 8,00 dan nilai tertinggi 15,80 (Tabel 4.1). Berdasarkan nilai *brix* tersebut (Gambar 4.5), diketahui bahwa aksesori yang dihasilkan dari proses fusi protoplasma memiliki nilai *brix* yang beragam.

Tabel 4.1 Deskripsi karakteristik data numerik

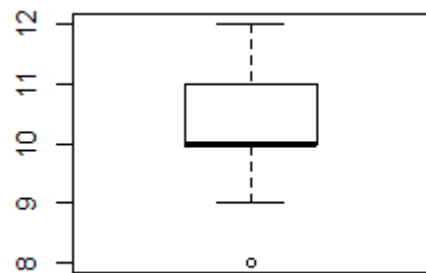
Variabel		Minimal	Maksimal	Median	Rata-rata	Standar Deviasi
Diameter Buah	(mm)	47,16	69,91	58,16	57,30	6,65
Tebal Kulit	(mm)	1,26	2,54	1,88	1,83	0,31
Jumlah Juring	(buah)	8,00	12,00	10,00	10,00	0,86
Tebal Daging Kulit	(mm)	1,21	2,62	1,61	1,72	0,35
Diameter <i>Axis</i>	(mm)	6,63	18,53	13,08	12,84	3,10
Jumlah Biji	(buah)	4,00	22,00	15,00	14,00	5,32
Berat Buah	(gram)	58,05	165,99	105,09	100,30	30,51
<i>Brix</i>	(derajat)	8,00	15,80	11,20	11,80	1,98



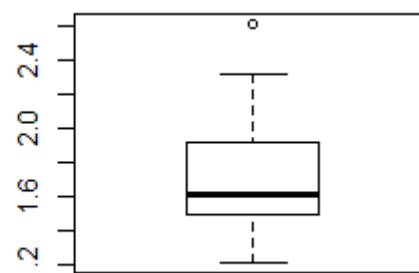
(a)



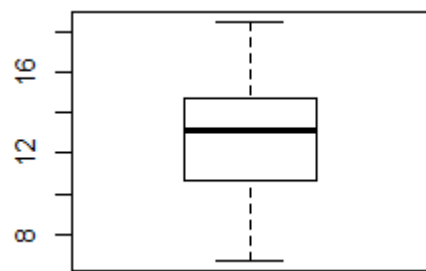
(b)



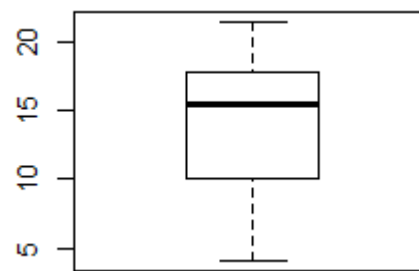
(c)



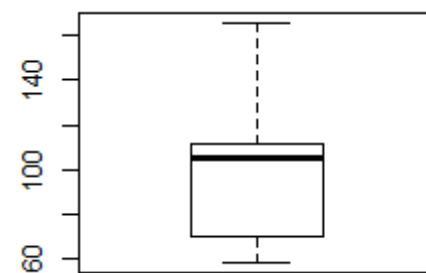
(d)



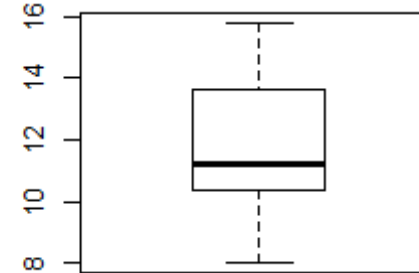
(e)



(f)



(g)



(h)

Gambar 4.5 Gambaran karakteristik aksesori jeruk berdasarkan (a) diameter buah, (b) tebal kulit, (c) jumlah juring, (d) tebal daging kulit, (e) diameter *axis*, (f) jumlah biji, (g) berat buah, dan (h) *brix*.

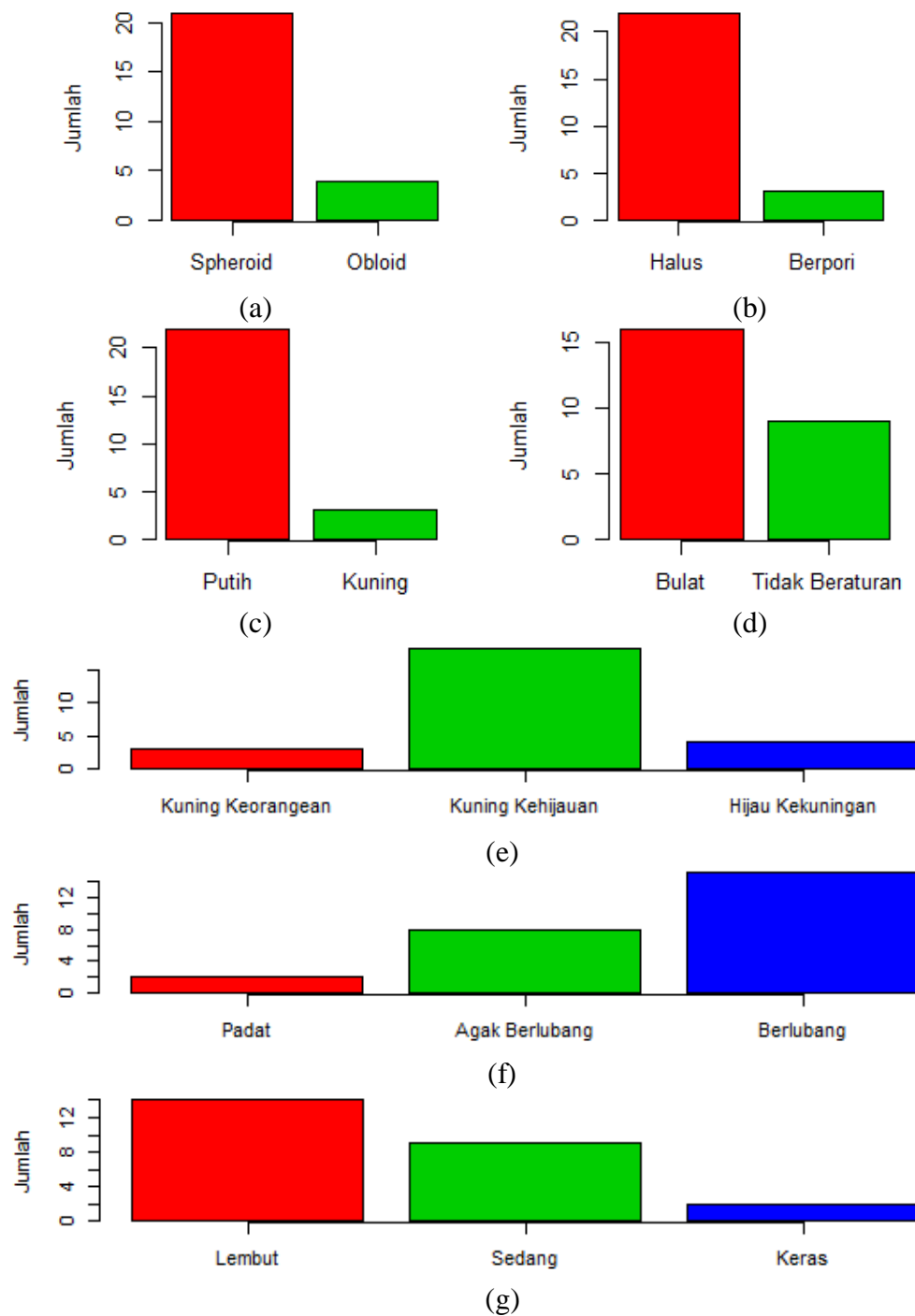
4.3.1.2 Data Kategorik

Data kategorik pada karakteristik morfologi jeruk meliputi data-data dari variabel bentuk buah, warna kulit, permukaan kulit, bentuk *axis* buah, warna daging kulit, tampak *axis*, dan tekstur *pulp*. Karakteristik bentuk buah dari 25 objek pengamatan yang diperoleh adalah 21 buah atau 84% berbentuk sferoid dan 4 buah atau 16% berbentuk obloid (Gambar 4.6). Hal ini menunjukkan bahwa bentuk buah yang dihasilkan hanya memiliki dua kategori dengan kategori sferoid sangat dominan dibandingkan kategori obloid.

Karakteristik warna kulit dari 25 objek pengamatan yang diperoleh adalah 3 buah atau 12% berwarna kuning keorangean, 18 buah atau 72% berwarna kuning kehijauan, dan 4 buah atau 16% berwarna hijau kekuningan (Gambar 4.6). Hal ini menunjukkan bahwa warna kulit buah yang dihasilkan hanya memiliki tiga kategori dengan kategori kuning kehijauan yang dominan dibanding dua kategori lainnya.

Karakteristik permukaan kulit dari 25 objek pengamatan yang diperoleh adalah 22 buah atau 88% berpermukaan halus dan 3 buah atau 12% berpermukaan berpori (Gambar 4.6). Hal ini menunjukkan bahwa permukaan kulit buah yang dihasilkan hanya memiliki dua kategori dengan kategori permukaan halus yang sangat dominan dibandingkan kategori permukaan berpori. Karakteristik warna daging kulit dari 25 objek pengamatan yang diperoleh adalah 22 buah atau 88% berwarna putih, dan 3 buah atau 12% berwarna kuning (Gambar 4.6). Hal ini menunjukkan bahwa warna daging kulit buah yang dihasilkan hanya memiliki dua kategori dengan kategori warna putih yang dominan dibanding warna kuning.

Karakteristik *axis* buah dari 25 objek pengamatan yang diperoleh adalah 2 buah atau 8% berbentuk padat, 8 buah atau 32% agak berlubang, dan 15 buah atau 60% berlubang (Gambar 4.6). Hal ini menunjukkan bahwa *axis* buah yang dihasilkan memiliki tiga kategori dengan kategori bentuk *axis* berlubang yang dominan dibanding dua kategori lainnya. Karakteristik tampak *axis* dari 25 objek pengamatan yang diperoleh adalah 16 buah atau 64% berbentuk bulat dan 9 buah atau 36% berbentuk tidak beraturan (Gambar 4.6). Hal ini menunjukkan bahwa tampak *axis* buah yang dihasilkan memiliki dua kategori dengan kategori bulat yang lebih dominan dari pada kategori tidak beraturan.



Gambar 4.6 Gambaran karakteristik aksesori jeruk berdasarkan (a) bentuk buah, (b) permukaan kulit, (c) warna daging kulit, (d) tampak *axis*, (e) warna kulit, (f) *axis* buah, dan (g) tekstur *pulp*.

Pulp atau bulir jeruk merupakan bagian dalam daging jeruk yang banyak mengandung sari buah serta berbentuk seperti tetesan air. Tekstur dari *pulp* itu sendiri sangat mempengaruhi kualitas buah jeruk itu sendiri. Buah jeruk yang disukai oleh konsumen adalah buah jeruk dengan *pulp* yang banyak mengandung sari serta memiliki teskstur yang lembut. Karakteristik *pulp* dari 25 objek pengamatan yang diperoleh adalah 14 buah atau 56% bertekstur lembut, 9 buah atau 36% bertekstur sedang, dan 2 buah atau 8% bertekstur keras (Gambar 4.6). Hal ini menunjukkan bahwa tekstur *pulp* yang dihasilkan memiliki tiga kategori dengan kategori bertekstur lembut lebih dominan dibandingkan dua kategori lainnya.

4.3.2 Pengelompokan untuk Data Numerik

Pengelompokan untuk data numerik dilakukan dengan menggunakan metode hirarki *agglomerative*. Tahap pertama yang dilakukan dalam metode hirarki *agglomerative* adalah menyatakan (inisialisasi) setiap objek pengamatan sebagai suatu kelompok dengan anggota tunggal. Tahap berikutnya adalah membentuk matrik jarak antar objek pengamatan. Jarak yang digunakan dalam penelitian ini adalah jarak *euclidean* yang dihitung menggunakan persamaan 2.3. Jarak yang diperoleh dari 25 objek pengamatan tersebut dinyatakan dalam matriks **d** yang berukuran 25 x 25.

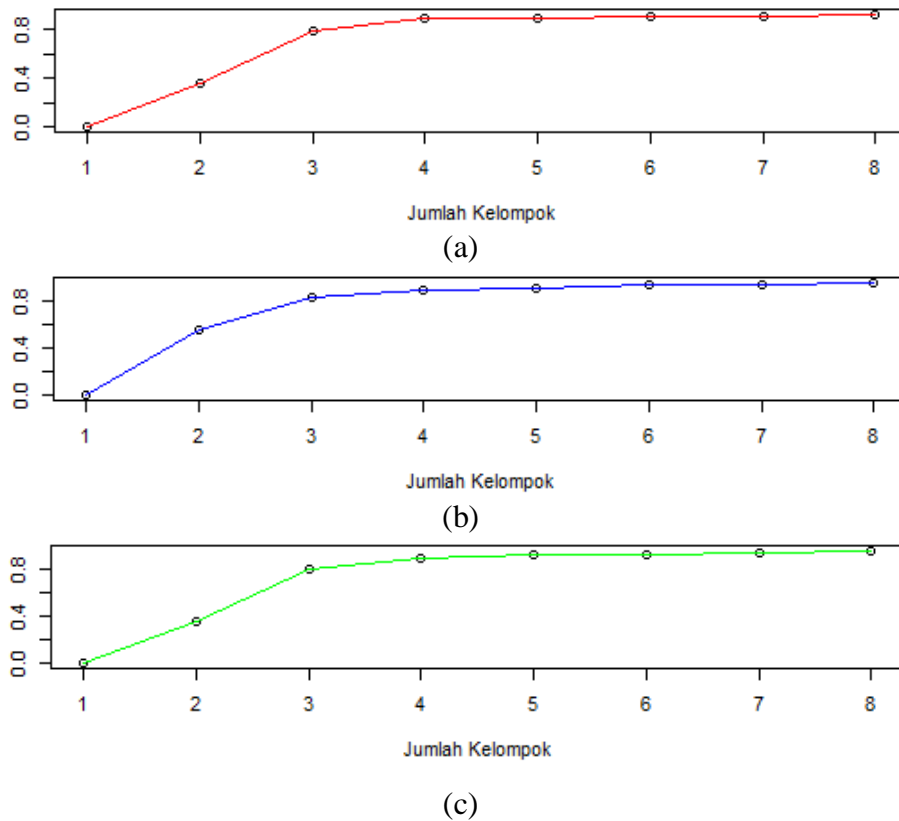
$$\mathbf{d} = \begin{bmatrix} 1.00 & 29.81 & 14.05 & 68.04 & 15.96 & \cdots & 21.84 & 22.49 & 22.92 & 36.29 & 34.08 \\ 29.81 & 1.00 & 26.10 & 94.46 & 45.17 & \cdots & 42.43 & 12.58 & 46.10 & 17.20 & 16.99 \\ 14.05 & 26.10 & 1.00 & 68.62 & 26.72 & \cdots & 17.10 & 23.87 & 24.63 & 31.93 & 29.70 \\ 68.04 & 94.46 & 68.62 & 1.00 & 57.65 & \cdots & 53.20 & 89.33 & 55.31 & 97.34 & 94.82 \\ 15.96 & 45.17 & 26.72 & 57.65 & 1.00 & \cdots & 23.58 & 36.24 & 19.47 & 49.18 & 46.92 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 21.84 & 42.43 & 17.10 & 53.20 & 23.58 & \cdots & 1.00 & 38.24 & 15.16 & 44.46 & 41.94 \\ 22.49 & 12.58 & 23.87 & 89.33 & 36.24 & \cdots & 38.24 & 1.00 & 38.34 & 17.96 & 16.75 \\ 22.92 & 46.10 & 24.63 & 55.31 & 19.47 & \cdots & 15.16 & 38.34 & 1.00 & 45.10 & 42.49 \\ 36.29 & 17.20 & 31.93 & 97.34 & 49.18 & \cdots & 44.46 & 17.96 & 45.10 & 1.00 & 3.43 \\ 34.08 & 16.99 & 29.70 & 94.82 & 46.92 & \cdots & 41.94 & 16.75 & 42.49 & 3.43 & 1.00 \end{bmatrix}$$

Matriks **d** merupakan matriks yang berisikan jarak dari seluruh kombinasi objek pengamatan dengan diagonal matriks bernilai 1 (jarak objek pengamatan dengan dirinya sendiri). Sebagai contoh, untuk nilai pada baris kedua kolom pertama matriks **d** tersebut menunjukkan bahwa jarak antara pengamatan pertama dengan pengamatan kedua adalah sebesar 29,81. Setelah diperoleh nilai jarak antar objek pengamatan, selanjutnya dilakukan pengabungan kelompok dengan jarak terdekat dan perbaharui matriks jarak dengan beberapa teknik pengelompokan antara kelompok antara lain *single linkage*, *complete linkage*, dan *average linkage*. Jumlah kelompok yang dibentuk berada pada nilai dua sampai delapan kelompok yang dilakukan dengan *syntax* pada Lampiran 14. Hasil pengelompokan data numerik menggunakan *software R* dapat disajikan pada Lampiran 22.

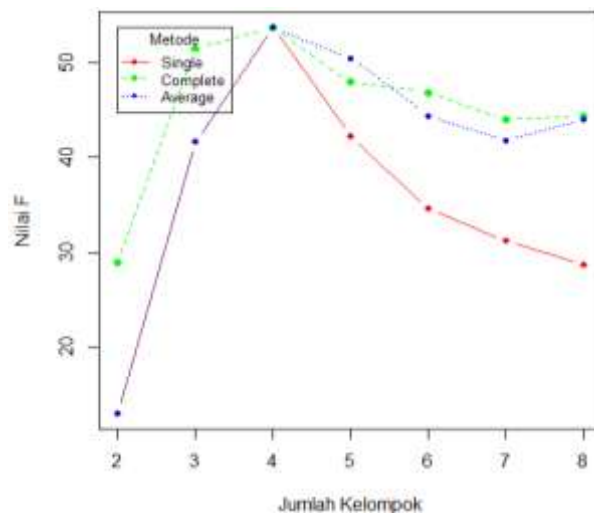
Setelah diperoleh hasil pengelompokan, tahap berikutnya adalah menghitung indeks validitas kelompok (nilai *R-Square* dan nilai *F*). Untuk melakukan penaksiran jumlah kelompok optimum adalah dengan membentuk plot dari indeks validitas kelompok tersebut. Pada indeks validitas berupa nilai *R-Square*, jumlah kelompok optimum merupakan jumlah kelompok dengan nilai yang meningkat sangat tajam dan menjadi cenderung stabil untuk jumlah kelompok berikutnya. Pada indeks validitas berupa nilai *F*, jumlah kelompok optimum merupakan jumlah kelompok dengan nilai *F* tertinggi dimana nilai *F* menurun pada jumlah kelompok berikutnya.

Gambar 4.7 menunjukkan bahwa taksiran jumlah kelompok optimum adalah 4 kelompok untuk metode *single linkage*, *complete linkage*, maupun *average linkage*. Hal ini dikarenakan pada jumlah kelompok sebanyak 4 kelompok masih mengalami perubahan nilai *R-Square* yang cukup besar dari nilai *R-Square* untuk 3 kelompok dan mengalami perubahan nilai *R-Square* yang kecil untuk 5 kelompok dan seterusnya. Selain menggunakan nilai *R-Square*, penentuan jumlah kelompok optimum juga dapat diketahui dari nilai *F*. Gambar 4.8 menunjukkan bahwa taksiran jumlah kelompok optimum adalah 4 kelompok untuk metode *single linkage*, *complete linkage*, maupun *average linkage*. Hal ini dikarenakan pada jumlah kelompok sebanyak 4 kelompok merupakan nilai *F* tertinggi dan mengalami

penurunan nilai F untuk 5 kelompok dan seterusnya. Nilai tersebut merupakan perpotongan antara ketiga metode, sehingga semakin menunjukkan bahwa jumlah kelompok optimum untuk pengelompokan data numerik adalah 4 kelompok.



Gambar 4.7 Rangkuman nilai R -Square berdasarkan jumlah kelompok untuk metode (a) *single linkage*, (b) *complete linkage*, dan (c) *single linkage*.

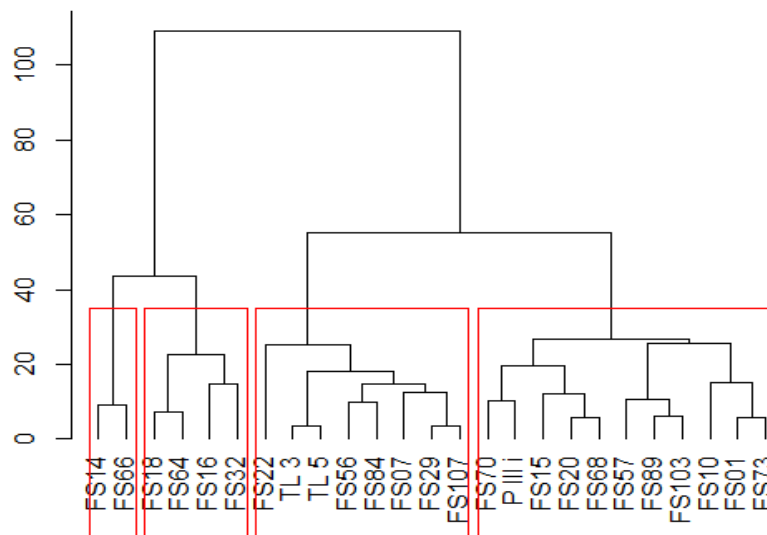


Gambar 4.8 Rangkuman nilai F berdasarkan jumlah kelompok.

Setelah memperoleh jumlah kelompok optimum, selanjutnya dipilih metode teknik pengelompokan terbaik untuk data dengan melihat nilai ratio S_W dan S_B dari masing-masing metode. Nilai ratio yang terbentuk disajikan pada Tabel 4.4. Dengan menggunakan rumusan pada persamaan 2.15 dan 2.16, diperoleh nilai ratio untuk setiap metode adalah sama yaitu sebesar 0,6688 (Tabel 4.2). Nilai ini menunjukkan bahwa simpangan baku dalam kelompok bernilai 0,6688 kali dari simpangan baku antar kelompok. Nilai ratio yang sama untuk ketiga metode disebabkan karena ketiga metode memberikan hasil pengelompokan yang sama. Hal menunjukkan bahwa pengelompokan data numerik dengan 4 kelompok merupakan pengelompokan yang tepat untuk metode hirarki *agglomerative*.

Tabel 4.2 Nilai ratio hasil pengelompokan data numerik dengan 4 kelompok

Metode	Sw	Sb	Ratio
<i>Single Linkage</i>	1,0516	1,5723	0,6688
<i>Complete Linkage</i>	1,0516	1,5723	0,6688
<i>Average Linkage</i>	1,0516	1,5723	0,6688



Gambar 4.9 Dendrogram hasil pengelompokan data numerik.

Gambar 4.9 menunjukkan hasil pengelompokan data numerik menggunakan metode hirarki *agglomerative* dengan jarak *euclidean* dalam 4 kelompok dalam

bentuk *dendogram*. Karakteristik hasil pengelompokan yang diperoleh adalah sebagai berikut,

a. Kelompok 1

Kelompok 1 merupakan kelompok yang beranggotakan 11 objek pengamatan yaitu FS70, PIIIi, FS15, FS20, FS68, FS57, FS89, FS103, FS10, FS01, dan FS73. Kelompok ini memiliki nilai rata-rata diameter buah 58,23 mm, tebal kulit 1,85 mm, jumlah juring sebanyak 10 buah, tebal daging kulit 1,83 mm, diameter *axis* 13,24 mm, jumlah biji sebanyak 15 buah, berat 103 gram, dan nilai *brix* sebesar 11,35.

b. Kelompok 2

Kelompok 2 merupakan kelompok yang beranggotakan 8 objek pengamatan yaitu FS22, TL3, TL5, FS56, FS84, FS07, FS29, dan FS107. Kelompok ini memiliki nilai rata-rata diameter buah 57,28 mm, tebal kulit 1,86 mm, jumlah juring sebanyak 11 buah, tebal daging kulit 1,62 mm, diameter *axis* 12,70 mm, jumlah biji sebanyak 13 buah, berat 65,51 gram, dan nilai *brix* sebesar 11.38.

c. Kelompok 3

Kelompok 3 merupakan kelompok yang beranggotakan 2 objek pengamatan yaitu FS14 dan FS66. Kelompok ini memiliki nilai rata-rata diameter buah 53,06 mm, tebal kulit 1,99 mm, jumlah juring sebanyak 10 buah, tebal daging kulit 1,89 mm, diameter *axis* 13,27 mm, jumlah biji sebanyak 16 buah, 163,72 gram, dan nilai *brix* sebesar 12,00.

d. Kelompok 4

Kelompok 4 merupakan kelompok yang beranggotakan 4 objek pengamatan yaitu FS18, FS64, FS16, dan FS32. Kelompok ini memiliki nilai rata-rata diameter buah 56,06 mm, tebal kulit 1,63 mm, jumlah juring sebanyak 11 buah, tebal daging kulit 1,52 mm, diameter *axis* 11,79 mm, jumlah biji sebanyak 14 buah, berat 130.77 gram, dan nilai *brix* sebesar 12,70.

4.3.3 Pengelompokan untuk Data Kategorik

Pengelompokan untuk data numerik dilakukan dengan menggunakan metode ROCK. Tahap pertama yang dilakukan dalam metode ROCK adalah menyatakan (inisialisasi) setiap objek pengamatan sebagai suatu kelompok dengan anggota tunggal. Tahap berikutnya adalah membentuk matrik jarak antar objek pengamatan dengan menggunakan rumusan pada persamaan 2.11. Jarak yang diperoleh dari 25 objek pengamatan tersebut dinyatakan dalam matriks **sim** yang berukuran 25 x 25.

$$\mathbf{sim} = \begin{bmatrix} 1.00 & 0.30 & 0.10 & 0.39 & 0.04 & \dots & 0.14 & 0.08 & 0.14 & 0.08 & 0.14 \\ 0.30 & 1.00 & 0.05 & 0.30 & 0.00 & \dots & 0.08 & 0.14 & 0.08 & 0.04 & 0.08 \\ 0.10 & 0.05 & 1.00 & 0.30 & 0.00 & \dots & 0.00 & 0.04 & 0.00 & 0.05 & 0.00 \\ 0.39 & 0.30 & 0.30 & 1.00 & 0.04 & \dots & 0.05 & 0.00 & 0.05 & 0.00 & 0.05 \\ 0.04 & 0.00 & 0.00 & 0.04 & 1.00 & \dots & 0.00 & 0.03 & 0.08 & 0.04 & 0.08 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0.14 & 0.08 & 0.00 & 0.05 & 0.00 & \dots & 1.00 & 0.08 & 0.14 & 0.20 & 0.14 \\ 0.08 & 0.14 & 0.04 & 0.00 & 0.03 & \dots & 0.08 & 1.00 & 0.08 & 0.04 & 0.08 \\ 0.14 & 0.08 & 0.00 & 0.05 & 0.08 & \dots & 0.14 & 0.08 & 1.00 & 0.08 & 0.28 \\ 0.08 & 0.04 & 0.05 & 0.00 & 0.04 & \dots & 0.20 & 0.04 & 0.08 & 1.00 & 0.08 \\ 0.14 & 0.08 & 0.00 & 0.05 & 0.08 & \dots & 0.14 & 0.08 & 0.28 & 0.08 & 1.00 \end{bmatrix}$$

Matriks **sim** merupakan matriks yang berisikan jarak dari seluruh kombinasi objek pengamatan dengan diagonal matriks bernilai 1 (jarak objek pengamatan dengan dirinya sendiri). Sebagai contoh, untuk nilai pada baris kedua kolom pertama matriks **sim** tersebut menunjukkan bahwa jarak antara pengamatan pertama dengan pengamatan kedua adalah sebesar 0,30. Setelah diperoleh jarak antara pengamatan, selanjutnya ditentukan nilai θ sebagai batas penentuan tetangga. Informasi mengenai hubungan tetangga antara objek pengamatan dapat dinyatakan dengan matriks **A**. Matriks **A** merupakan matriks berukuran 25 x 25 yang bernilai 1 jika objek tersebut bertetangga dan bernilai 0 jika objek tersebut tidak bertetangga. Sebagai contoh untuk jarak antara pengamatan pertama dengan pengamatan kedua yang bernilai 0,30, maka dengan nilai $\theta = 0,17$ dapat dinyatakan bahwa pengamatan tersebut merupakan tetangga sehingga matriks **A** pada baris kedua kolom pertama bernilai 1.

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & \cdots & 0 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \cdots & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 1 & 0 & 1 \end{bmatrix}$$

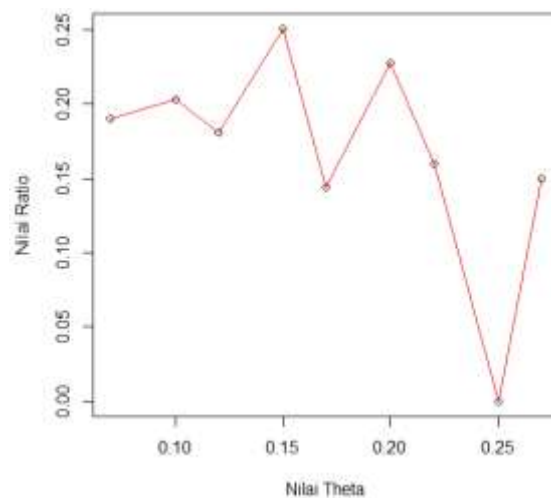
Setelah diperoleh informasi tetangga antar seluruh kombinasi pengamatan, selanjutnya dilakukan perhitungan jumlah *link* dan *goodness measure*. Perhitungan jumlah *link* dilakukan dengan melakukan perkalian matriks **A** dengan matriks **A** itu sendiri. Perhitungan jumlah *link* tersebut, dinyatakan dalam matriks **link** yang berukuran 25 x 25. Sebagai contoh, untuk nilai pada baris kedua kolom pertama matriks **link** tersebut menunjukkan bahwa jumlah *link* antara pengamatan pertama dengan pengamatan kedua adalah sebanyak 6 *link*.

$$\mathbf{link} = \begin{bmatrix} 11 & 6 & 4 & 5 & 0 & \cdots & 3 & 0 & 3 & 0 & 3 \\ 6 & 6 & 3 & 5 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 \\ 4 & 3 & 6 & 4 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 \\ 5 & 5 & 4 & 6 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & \cdots & 0 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 3 & 0 & 0 & 0 & 0 & \cdots & 6 & 0 & 3 & 2 & 3 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 2 & 0 & 0 & 0 \\ 3 & 0 & 0 & 0 & 0 & \cdots & 3 & 0 & 5 & 0 & 5 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 2 & 0 & 0 & 3 & 0 \\ 3 & 0 & 0 & 0 & 0 & \cdots & 3 & 0 & 5 & 0 & 5 \end{bmatrix}$$

Dalam penelitian ini digunakan beberapa nilai θ yaitu 0,07; 0,10; 0,12; 0,15; 0,17; 0,20; 0,22; 0,25; 0,27 dan 0,30. Nilai tersebut ditentukan oleh peneliti yang disesuaikan dengan jarak objek pengamatan dan hasil pengelompokan yang

diharapkan. Hasil yang diharapkan adalah hasil pengelompokan dimana semua objek pengamatan tidak berada dalam satu kelompok, serta tidak terdapat kelompok dengan anggota tunggal. Hasil pengelompokan metode ROCK menggunakan *software R* yang dilakukan dengan bantuan *package cba* disajikan pada Lampiran 23. Hasil pengelompokan terbaik ditentukan dari nilai ratio S_W dan S_B terkecil.

Gambar 4.10 menunjukkan bahwa nilai rasio terkecil merupakan pengelompokan dengan θ sebesar 0,25. Akan tetapi hasil tersebut tidak dapat digunakan, dikarenakan tidak sesuai dengan hasil yang diharapkan dimana seluruh objek pengamatan berada dalam satu kelompok yang sama. Oleh karena itu, hasil pengelompokan terbaik merupakan pengamatan dengan nilai rasio terkecil kedua yaitu pengelompokan dengan nilai θ sebesar 0,17 dengan ratio bernilai 0,1442. Nilai tersebut menunjukkan bahwa simpangan baku dalam kelompok bernilai 0,1442 kali dari simpangan baku antar kelompok. Dengan kata lain, variansi data dalam kelompok memberikan nilai simpangan lebih kecil dibandingkan variansi antar kelompok.



Gambar 4.10 Nilai ratio pengelompokan data kategorik.

Hasil pengelompokan metode ROCK dengan nilai θ sebesar 0,17 menghasilkan 3 kelompok dengan anggota setiap kelompok ditunjukkan pada Tabel 4.3. Karakteristik hasil pengelompokan metode ROCK yang diperoleh adalah sebagai berikut,

a. Kelompok 1

Kelompok 1 merupakan kelompok yang beranggotakan 7 objek pengamatan yaitu FS14, FS29, FS32, FS57, FS73, PIIIi, dan TL5. Kelompok ini memiliki bentuk buah yang dominan sferoid (85%), warna kulit yang beragam, permukaan kulit yang dominan halus (71%), *axis* buah yang dominan berlubang (71%), warna daging kulit yang dominan putih (85%), *axis* yang tampak dominan bulat (71%), serta tekstur *pulp* yang beragam.

b. Kelompok 2

Kelompok 2 merupakan kelompok yang beranggotakan 3 objek pengamatan yaitu FS56, FS103, dan TL3. Kelompok ini memiliki bentuk buah sferoid, kulit berwarna kuning kehijauan, permukaan kulit yang halus, *axis* buah yang agak berlubang, warna daging kulit yang putih, *axis* yang tampak bulat, serta tekstur *pulp* yang lembut.

c. Kelompok 3

Kelompok 3 merupakan kelompok yang beranggotakan 15 objek pengamatan. Kelompok ini memiliki bentuk buah yang dominan sferoid (87%), warna kulit yang beragam, permukaan kulit yang dominan halus (87%), *axis* buah yang dominan berlubang (80%), warna daging kulit yang dominan putih (93%), *axis* yang tampak beragam, serta tekstur *pulp* yang beragam.

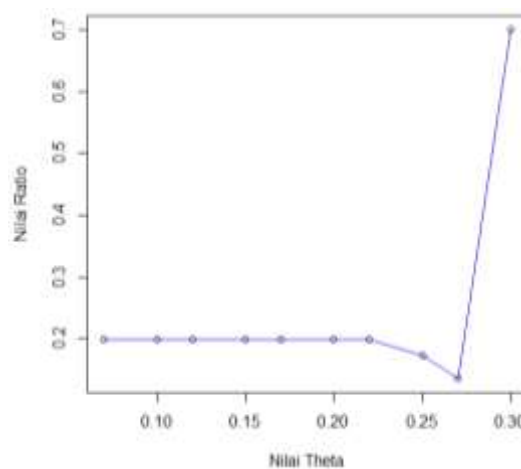
Tabel 4.3 Hasil pengelompokan data kategorik dengan nilai $\theta = 0,17$

Kelompok			
1	2	3	
FS57	FS103	FS01	FS22
FS73	FS56	FS10	FS84
P III i	TL 3	FS15	FS107
FS29		FS20	FS66
TL 5		FS68	FS16
FS14		FS70	FS18
FS32		FS89	FS64
		FS07	

4.3.4 Pengelompokan Data Campuran dengan Metode Ensemble ROCK

Tahapan pertama dalam melakukan analisis pengelompokan ensemble ROCK untuk data campuran adalah dengan melakukan pengelompokan masing-masing jenis data menggunakan metodenya masing-masing. Hasil pengelompokan untuk data numerik yang diperoleh dari analisis pada subbab 4.3.2 dinyatakan sebagai *output* 1, serta hasil pengelompokan untuk data kategorik yang diperoleh dari analisis pada subbab 4.3.3 dinyatakan sebagai *output* 2. Berikutnya kedua hasil pengelompokan tersebut dinyatakan sebagai variabel kategorik (tahap ensemble) yang kemudian dikelompokkan dengan metode ROCK.

Dalam analisis ini digunakan beberapa nilai θ seperti pada pengelompokan data kategorik di subbab 4.4.3 yaitu 0,07; 0,10; 0,12; 0,15; 0,17; 0,20; 0,22; 0,25; 0,27 dan 0,30. Hasil pengelompokan metode ensemble ROCK menggunakan *software R* yang dilakukan dengan bantuan *package cba* disajikan pada Lampiran 24. Hasil pengelompokan terbaik ditentukan dari nilai ratio S_W dan S_B terkecil.



Gambar 4.11 Nilai ratio pengelompokan ensemble ROCK.

Gambar 4.11 menunjukkan bahwa nilai rasio terkecil merupakan pengelompokan dengan θ sebesar 0,27 dengan ratio bernilai 0,1358. Nilai tersebut menunjukkan bahwa simpangan baku dalam kelompok bernilai 0,1358 kali dari simpangan baku antar kelompok. Dengan kata lain, variansi data dalam kelompok memberikan nilai simpangan lebih kecil dibandingkan variansi antar kelompok.

Hasil pengelompokan metode ensemble ROCK dengan nilai θ sebesar 0,27 menghasilkan 3 kelompok dengan anggota setiap kelompok ditunjukkan pada Tabel 4.4.

Tabel 4.4 Hasil pengelompokan ensemble ROCK dengan nilai $\theta = 0,27$

Kelompok				
1		2	3	
FS01	FS68	FS15	FS07	FS64
FS10	FS70	FS16	FS22	FS84
FS14	FS73	FS18	FS29	FS107
FS57	FS103	FS20	FS56	TL 5
FS66	P III i	FS32		
		FS89		
		TL 3		

Hasil pengelompokan yang diperoleh dari metode ensemble ROCK merupakan tiga kelompok dengan perbandingan karakteristik numerik ketiga kelompok disajikan pada Gambar 4.12. Berdasarkan diameter buah, ketiga kelompok memiliki rata-rata yang tidak jauh berbeda, akan tetapi kelompok 2 dan kelompok 3 memiliki variansi yang lebih besar dari pada kelompok 1. Berdasarkan tebal kulit, ketiga kelompok memiliki rata-rata yang berbeda serta variansi yang sangat berbeda dimana kelompok 1 memiliki variansi terbesar dan kelompok 3 dengan variansi terkecil. Berdasarkan jumlah juring, ketiga kelompok memiliki rata-rata yang sama, akan tetapi memiliki variansi yang sangat berbeda dimana kelompok 1 memiliki variansi terbesar dan kelompok 2 dengan variansi terkecil. Berdasarkan tebal daging kulit, kelompok 3 memiliki rata-rata serta variansi yang sangat berbeda daripada kelompok 1 dan kelompok 2. Berdasarkan diameter *axis*, ketiga kelompok memiliki rata-rata yang berbeda serta variansi kelompok 1 lebih kecil daripada kelompok 2 dan 3. Berdasarkan jumlah biji, ketiga kelompok memiliki rata-rata yang tidak jauh berbeda serta variansi kelompok 2 lebih kecil daripada kelompok 1 dan 3. Berdasarkan berat buah, ketiga kelompok memiliki rata-rata yang berbeda serta variansi yang sangat berbeda dimana kelompok 2 memiliki variansi terbesar dan kelompok 3 dengan variansi terkecil. Berdasarkan

nilai *brix*, ketiga kelompok memiliki rata-rata yang berbeda serta variansi yang sangat berbeda dimana kelompok 2 memiliki variansi terbesar dan kelompok 3 dengan variansi terkecil.

Karakteristik hasil pengelompokan metode ensemble ROCK yang diperoleh untuk masing-masing kelompok adalah sebagai berikut,

a. Kelompok 1

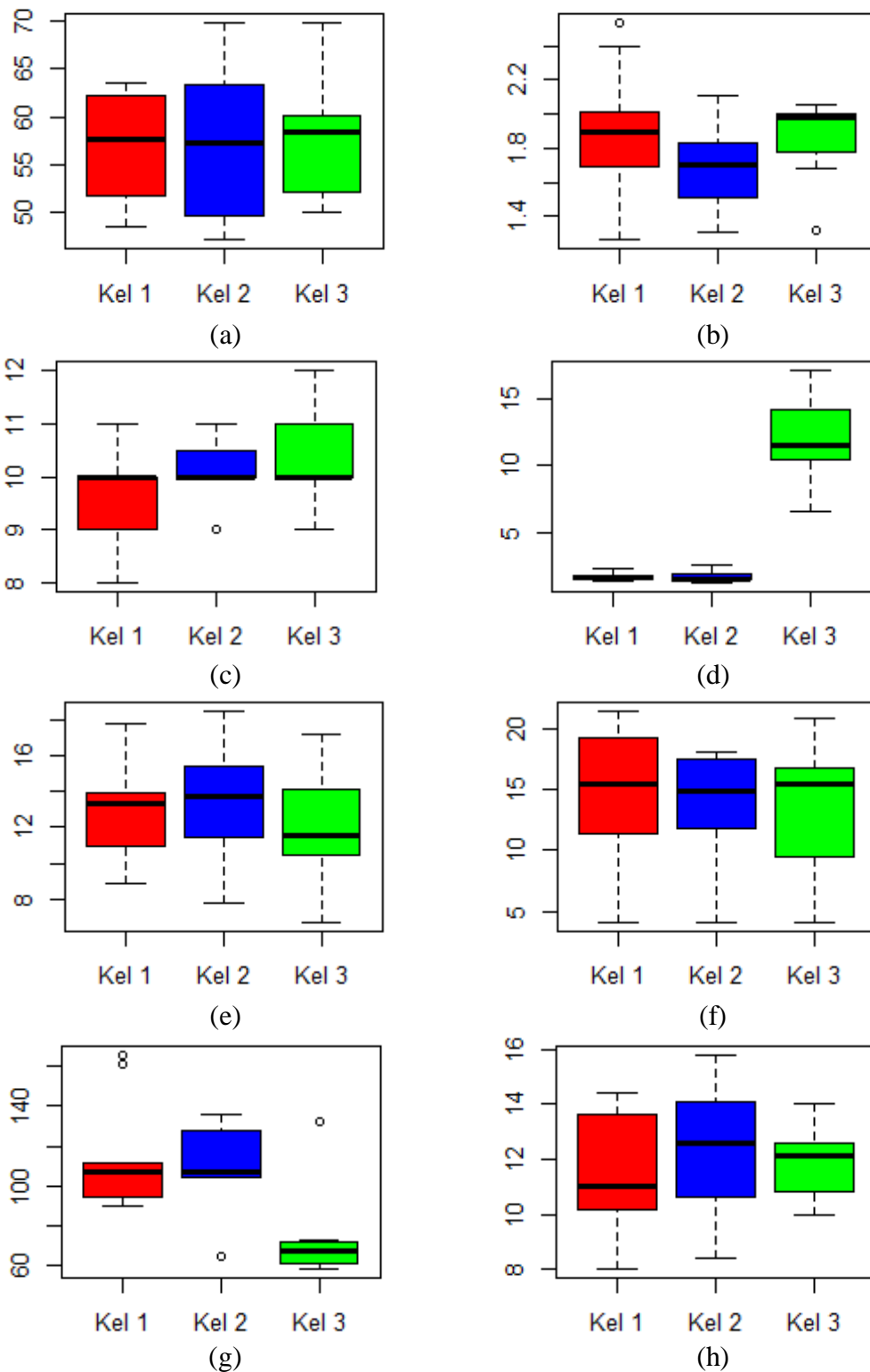
Kelompok 1 merupakan kelompok yang beranggotakan 10 objek pengamatan. Berdasarkan variabel numerik (Tabel 4.5), kelompok ini memiliki nilai rata-rata diameter buah 57,03 mm, tebal kulit 1,90 mm, jumlah juring sebanyak 10 buah, tebal daging kulit 1,78 mm, diameter *axis* 13,11 mm, jumlah biji sebanyak 15 buah, berat 114,40 gram, dan nilai *brix* sebesar 11,34. Berdasarkan variabel kategorik (Tabel 4.6), kelompok ini memiliki bentuk buah yang dominan sferoid, warna kulit yang dominan kuning kehijauan, permukaan kulit yang dominan halus, *axis* buah yang beragam, warna daging kulit yang putih, *axis* tampak beragam, serta tekstur *pulp* beragam.

b. Kelompok 2

Kelompok 2 merupakan kelompok yang beranggotakan 7 objek pengamatan. Berdasarkan variabel numerik (Tabel 4.5), kelompok ini memiliki nilai rata-rata diameter buah 57,24 mm, tebal kulit 1,69 mm, jumlah juring sebanyak 11 buah, tebal daging kulit 1,77 mm, diameter *axis* 13,38 mm, jumlah biji sebanyak 14 buah, berat 110,38 gram, dan nilai *brix* sebesar 12,31. Berdasarkan variabel kategorik (Tabel 4.6), kelompok ini memiliki bentuk buah yang dominan sferoid, warna kulit yang beragam, permukaan kulit yang beragam, *axis* buah yang beragam, warna daging kulit yang putih, *axis* yang tampak beragam, serta tekstur *pulp* yang beragam.

c. Kelompok 3

Kelompok 3 merupakan kelompok yang beranggotakan 8 objek pengamatan. Berdasarkan variabel numerik (Tabel 4.5), kelompok ini



Gambar 4.12 Gambaran perbandingan karakteristik hasil pengelompokan ensemble ROCK berdasarkan (a) diameter buah, (b) tebal kulit, (c) jumlah juring, (d) tebal daging kulit, (e) diameter *axis*, (f) jumlah biji, (g) berat buah, dan (h) *brix*.

memiliki nilai rata-rata diameter buah 57,68 mm, tebal kulit 1,86 mm, jumlah juring sebanyak 11 buah, tebal daging kulit 1,59 mm, diameter *axis* 12,01 mm, jumlah biji sebanyak 14 buah, berat 73,86 gram, dan nilai *brix* sebesar 11,88. Berdasarkan variabel kategorik (Tabel 4.6), kelompok ini memiliki bentuk buah yang dominan sferoid, warna kulit dominan kuning kehijauan, permukaan kulit halus, *axis* buah yang beragam, warna daging kulit dominan putih, *axis* yang tampak dominan bulat, serta tekstur *pulp* yang dominan lembut.

Tabel 4.5 Karakteristik variabel numerik hasil pengelompokan ensemble ROCK

Variabel		Kelompok 1		Kelompok 2		Kelompok 3	
Diameter Buah	(mm)	57,03	Terendah	57,24		57,68	Tertinggi
Tebal Kulit	(mm)	1,90	Tertinggi	1,69	Terendah	1,86	
Jumlah Juring	(buah)	10,00	Terendah	11,00		11,00	
Tebal Daging Kulit	(mm)	1,78	Tertinggi	1,77		1,59	Terendah
Diamter <i>Axis</i>	(mm)	13,11		13,38	Tertinggi	12,01	Terendah
Jumlah Biji	(buah)	15,00	Tertinggi	14,00		14,00	
Berat Buah	(gram)	114,40	Tertinggi	110,38		73,86	Terendah
<i>Brix</i>	(derajat)	11,34	Terendah	12,31	Tertinggi	11,88	

Tabel 4.6 Karakteristik variabel kategorik hasil pengelompokan ensemble ROCK

Variabel	Kelompok 1		Kelompok 2		Kelompok 3	
Bentuk Buah	1	(90%)	1	(86%)	1	(75%)
Warna Kulit	4	(80%)	4	(57%)	4	(75%)
Permukaan Kulit	1	(80%)	1	(71%)	1	(100%)
<i>Axis</i> Buah	3	(60%)	3	(57%)	3	(63%)
Warna Daging Kulit	1	(90%)	1	(100%)	1	(75%)
Tampak <i>Axis</i>	1	(60%)	2	(57%)	1	(88%)
Tekstur <i>Pulp</i>	2	(60%)	1	(71%)	1	(75%)

4.3.5 Pengelompokan Data Campuran dengan Metode Ensemble SWFM

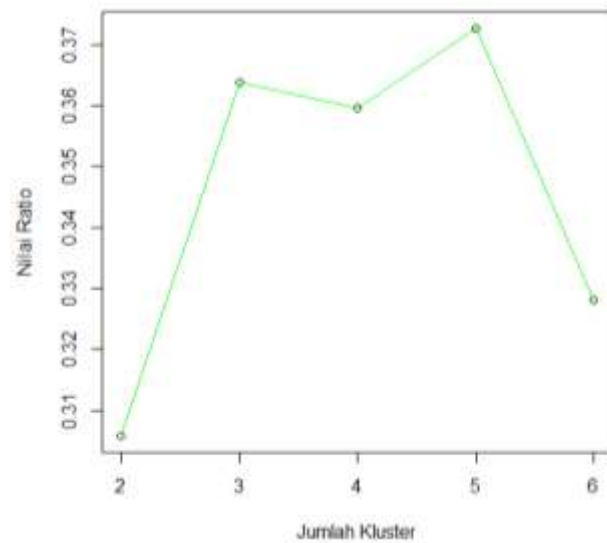
Tahapan pertama dalam melakukan analisis pengelompokan ensemble SWFM untuk data campuran adalah dengan melakukan pengelompokan masing-masing jenis data menggunakan metodenya masing-masing. Hasil pengelompokan

untuk data numerik yang diperoleh dari analisis pada subbab 4.3.2 dinyatakan sebagai *output* 1, serta hasil pengelompokan untuk data kategorik yang diperoleh dari analisis pada subbab 4.3.3 dinyatakan sebagai *output* 2. Berikutnya kedua hasil pengelompokan tersebut dinyatakan sebagai variabel kategorik (tahap ensemble) yang kemudian dikelompokkan dengan metode SWFM.

Tahap berikutnya adalah membentuk matrik jarak antar objek pengamatan. Jarak yang digunakan dalam penelitian ini adalah jarak *similarity weight* yang dihitung menggunakan persamaan 2.13, selanjutnya dibentuk nilai jarak F antar objek pengamatan berdasarkan persamaan 2.14. Setelah diperoleh jarak F antar objek pengamatan, kemudian dilakukan pengabungan kelompok dengan jarak terdekat dan perbaharui matriks jarak. Jumlah kelompok yang dibentuk berada pada nilai dua sampai enam kelompok yang dilakukan dengan *syntax* yang dilampirkan pada Lampiran 20. Hasil pengelompokan ensemble SWFM menggunakan *software R* disajikan pada Lampiran 25.

Hasil pengelompokan terbaik atau jumlah kelompok optimum ditentukan dari nilai ratio S_W dan S_B terkecil. Gambar 4.13 menunjukkan bahwa nilai rasio terkecil merupakan pengelompokan dengan jumlah kelompok sebanyak 2 kelompok dengan ratio bernilai 0,3059. Nilai tersebut menunjukkan bahwa simpangan baku dalam kelompok bernilai 0,3059 kali dari simpangan baku antar kelompok. Dengan kata lain, variansi data dalam kelompok memberikan nilai simpangan lebih kecil dibandingkan variansi antar kelompok. Hasil pengelompokan metode ensemble SWFM terbaik menghasilkan 2 kelompok dengan anggota setiap kelompok disajikan pada Tabel 4.7.

Hasil pengelompokan yang diperoleh dari metode ensemble SWFM merupakan dua kelompok dengan perbandingan karakteristik numerik kedua kelompok disajikan pada Gambar 4.14. Berdasarkan diameter buah, kedua kelompok memiliki rata-rata yang sama, akan tetapi kelompok 1 memiliki variansi yang lebih besar dari pada kelompok 2.



Gambar 4.13 Nilai ratio pengelompokan ensemble SWFM.

Tabel 4.7 Hasil pengelompokan ensemble SWFM dengan 2 kelompok

Kelompok		
	1	2
FS01	FS56	FS73
FS07	FS57	FS84
FS10	FS64	TL 3
FS14	FS66	
FS15	FS68	
FS16	FS70	
FS18	FS89	
FS20	FS103	
FS22	FS107	
FS29	P III i	
FS32	TL 5	

Berdasarkan tebal kulit, kedua kelompok memiliki rata-rata yang berbeda dengan variansi yang tidak jauh berbeda. Berdasarkan jumlah juring, kedua kelompok memiliki rata-rata sangat berbeda, akan tetapi memiliki variansi yang tidak jauh berbeda. Berdasarkan tebal daging kulit, kedua kelompok memiliki rata-rata yang sama, akan tetapi kelompok 1 memiliki variansi yang lebih besar daripada kelompok 2. Berdasarkan diameter *axis*, kedua kelompok memiliki rata-rata yang

tidak jauh berbeda dengan variansi kelompok 1 lebih kecil daripada kelompok 2. Berdasarkan jumlah biji, kedua kelompok memiliki rata-rata yang sama, akan tetapi kelompok 1 memiliki variansi yang lebih besar daripada kelompok 2. Berdasarkan berat buah, kedua kelompok memiliki rata-rata yang sama, akan tetapi kelompok 1 memiliki variansi yang lebih kecil daripada kelompok 2. Berdasarkan nilai *brix*, kedua kelompok memiliki rata-rata yang sangat berbeda serta kelompok 1 memiliki variansi yang lebih kecil daripada kelompok 2.

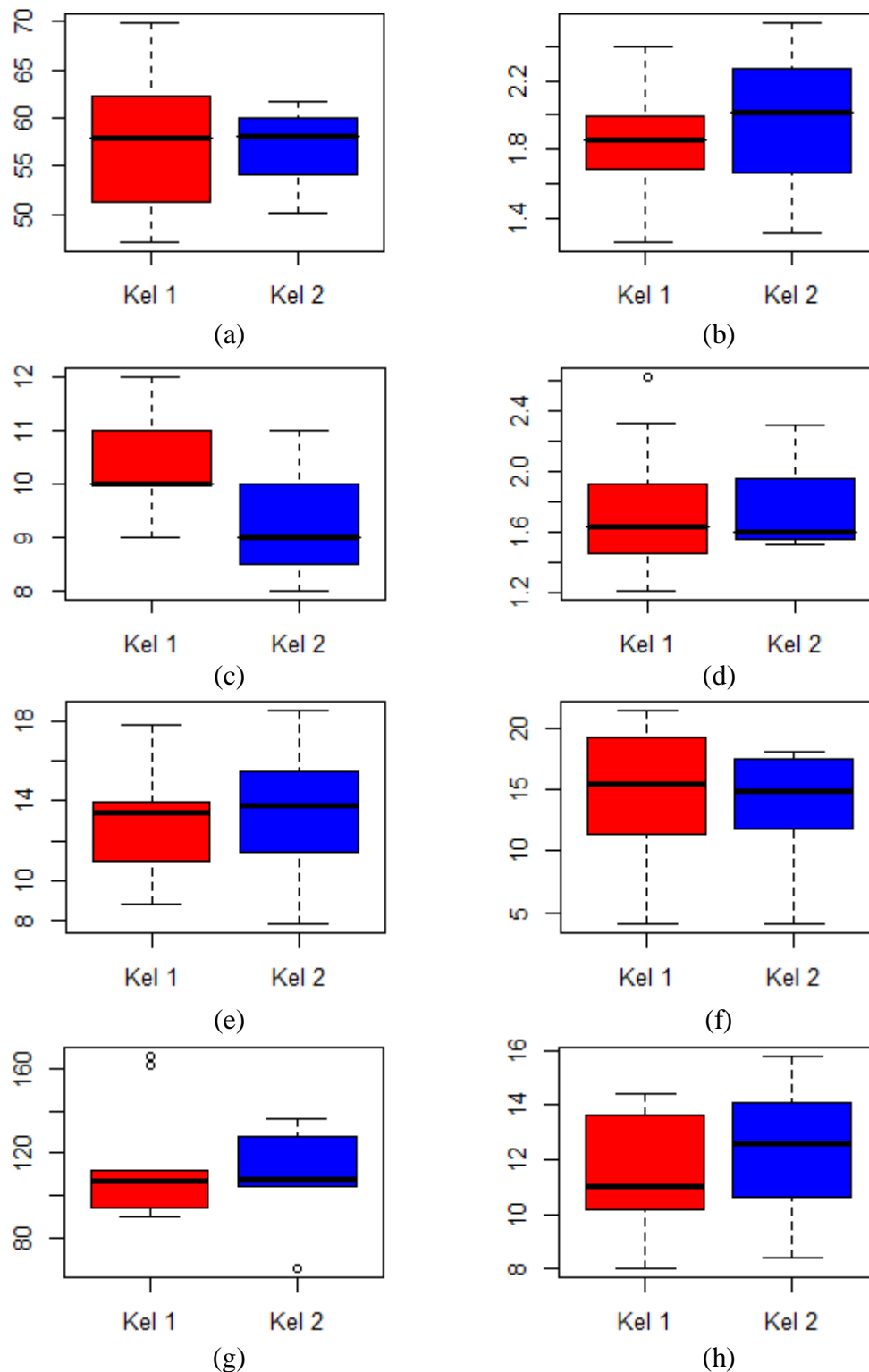
Karakteristik hasil pengelompokan metode ensemble SWFM yang diperoleh untuk masing-masing kelompok adalah sebagai berikut,

a. Kelompok 1

Kelompok 1 merupakan kelompok yang beranggotakan 22 objek pengamatan. Berdasarkan variabel numerik (Tabel 4.8), kelompok ini memiliki nilai rata-rata diameter buah 57,38 mm, tebal kulit 1,81 mm, jumlah juring sebanyak 11 buah, tebal daging kulit 1,70 mm, diameter *axis* 12,88 mm, jumlah biji sebanyak 15 buah, berat 104,09 gram, dan nilai *brix* sebesar 11,91. Berdasarkan variabel kategorik (Tabel 4.9), kelompok ini memiliki bentuk buah yang dominan spheroid, warna kulit yang dominan kuning kehijauan, permukaan kulit yang dominan halus, *axis* buah yang beragam, warna daging kulit yang dominan putih, *axis* tampak beragam, serta tekstur *pulp* beragam.

b. Kelompok 2

Kelompok 2 merupakan kelompok yang beranggotakan 3 objek pengamatan. Berdasarkan variabel numerik (Tabel 4.8), kelompok ini memiliki nilai rata-rata diameter buah 56,65 mm, tebal kulit 1,95 mm, jumlah juring sebanyak 10 buah, tebal daging kulit 1,81 mm, diameter *axis* 12,53 mm, jumlah biji sebanyak 12 buah, berat 72,53 gram, dan nilai *brix* sebesar 10,87. Berdasarkan variabel kategorik (Tabel 4.9), kelompok ini memiliki bentuk buah yang spheroid, warna kulit yang beragam, permukaan kulit yang dominan halus, *axis* buah yang dominan padat, warna daging kulit yang putih, *axis* yang tampak dominan tidak beraturan, serta tekstur *pulp* yang dominan lembut.



Gambar 4.14 Gambaran perbandingan karakteristik hasil pengelompokan ensemble SWFM berdasarkan (a) diameter buah, (b) tebal kulit, (c) jumlah juring, (d) tebal daging kulit, (e) diameter *axis*, (f) jumlah biji, (g) berat buah, dan (h) *brix*.

Tabel 4.8 Karakteristik variabel numerik hasil pengelompokan ensemble SWFM

Variabel		Kelompok 1		Kelompok 2	
Diameter Buah	(mm)	57,38	Tinggi	56,65	
Tebal Kulit	(mm)	1,81		1,95	Tinggi
Jumlah Juring	(buah)	11,00	Tinggi	10,00	
Tebal Daging Kulit	(mm)	1,70		1,81	Tinggi
Diamter <i>Axis</i>	(mm)	12,88	Tinggi	12,53	
Jumlah Biji	(buah)	15,00	Tinggi	12,00	
Berat Buah	(gram)	104,09	Tinggi	72,53	
<i>Brix</i>	(derajat)	11,91	Tinggi	10,87	

Tabel 4.9 Karakteristik variabel kategorik hasil pengelompokan ensemble SWFM

Variabel	Kelompok 1		Kelompok 2	
Bentuk Buah	1	(82%)	1	(100%)
Warna Kulit	4	(77%)	3	(33%)
Permukaan Kulit	1	(82%)	1	(67%)
<i>Axis</i> Buah	3	(68%)	2	(67%)
Warna Daging Kulit	1	(86%)	1	(100%)
Tampak <i>Axis</i>	1	(68%)	2	(67%)
Tekstur <i>Pulp</i>	1	(55%)	1	(67%)

4.4 Perbandingan Antara Metode Ensemble ROCK dengan Metode Ensemble SWFM

Pada subbab ini dijelaskan mengenai perbandingan hasil pengelompokan ensemble ROCK dan ensemble SWFM yang dilakukan untuk menjawab tujuan keempat dalam penelitian. Analisis perbandingan antara kedua metode untuk data campuran dilakukan berdasarkan nilai ratio S_W dan S_B yang terbentuk. Analisis ini dilakukan dengan bantuan program aplikasi pada *software R Project* menggunakan *syntax* yang ditampilkan pada Lampiran 21.

Analisis perbandingan kedua metode dilakukan dengan menggunakan nilai ratio untuk pengelompokan ensemble ROCK (Lampiran 28 (a)) dan nilai ratio untuk pengelompokan ensemble SWFM (Lampiran 28 (b)). Metode yang lebih baik merupakan metode yang menghasilkan nilai ratio S_W dan S_B lebih kecil. Berdasarkan *output* dari *software R Project*, diketahui bahwa nilai ratio yang lebih

kecil dimiliki oleh metode ensemble ROCK dengan ratio sebesar 0,1358 (Gambar 4.15). Hal ini menunjukkan bahwa metode ensemble ROCK memberikan kinerja pengelompokan yang lebih baik dibandingkan metode ensemble SWFM dengan studi kasus pengelompokan aksesori jeruk hasil fusi protoplasma.

```
> Banding.Ratio(Ratio.Ensemble.ROCK, Ratio.SWFM)
[1] "Berdasarkan nilai Ratio SW dan SB"
[1] "metode analisis yang terbaik adalah"
[1] "Metode Ensemble ROCK"
[1] "dengan nilai ratio terkecil yang dihasilkan sebesar"
[1] 0.1357722
```

Gambar 4.15 Tampilan *output* hasil perbandingan nilai rasio

Setelah diperoleh hasil pengelompokan yang optimum, selanjutnya dilakukan pengujian perbedaan rata-rata dari kelompok yang diperoleh. Pengujian dilakukan untuk mengetahui apakah hasil pengelompokan telah memberikan kelompok-kelompok yang berbeda secara signifikan. Pengujian dilakukan dengan menggunakan metode MANOVA berdasarkan data numerik dari penelitian. Sebelum dilakukan analisis dengan metode MANOVA, dilakukan dahulu pengujian normalitas multivariat dari data. Pengujian normalitas dilakukan dengan melakukan *mardia test* atau pengujian terhadap nilai nilai skewnes dan kurtosis. Rumusan hipotesis yang digunakan untuk pengujian nilai skewnes adalah sebagai berikut,

$$H_0 : \beta_{1,m_{numerik}} = 0 \text{ (data berdistribusi normal multivariat).}$$

$$H_1 : \beta_{1,m_{numerik}} \neq 0 \text{ (data tidak berdistribusi normal multivariat).}$$

Nilai statistik uji skewnes yang diperoleh adalah sebesar 105,3886 dengan nilai taksiran $b_{1,m_{numerik}}$ sebesar 25,2933, dan *p-value* sebesar 0,8266 (Lampiran 29). Dengan menggunakan taraf kepercayaan 90% atau nilai α sebesar 10%, diperoleh hasil bahwa *p-value* lebih besar dari α , sehingga dapat dinyatakan bahwa gagal tolak H_0 . Hal ini berarti bahwa berdasarkan nilai skewnes data dinyatakan berdistribusi normal multivariat. Selain nilai skewnes, dilakukan juga pengujian terhadap nilai kurtosis dengan hipotesis sebagai berikut,

$H_0 : \beta_{2,m_{numerik}} = 0$ (data berdistribusi normal multivariat).

$H_1 : \beta_{2,m_{numerik}} \neq 0$ (data tidak berdistribusi normal multivariat).

Nilai statistik uji kurtosis yang diperoleh adalah sebesar -1,0374 dengan nilai taksiran $b_{2,m_{numerik}}$ sebesar 74,7509, dan p -value sebesar 0,2995 (Lampiran 29). Dengan menggunakan taraf kepercayaan 90% atau nilai α sebesar 10%, diperoleh hasil bahwa p -value lebih besar dari α , sehingga dapat dinyatakan bahwa gagal tolak H_0 . Hal ini berarti bahwa berdasarkan nilai kurtosis data dinyatakan berdistribusi normal multivariat.

Setelah peroleh hasil bahwa data berdistribusi normal multivariat berdasarkan nilai skewnes dan kurtosis, maka data dapat digunakan pada analisis metode MANOVA. Rumusan hipotesis pada MANOVA yang digunakan adalah sebagai berikut,

$H_0 : \bar{\mu}_1 = \bar{\mu}_2 = \bar{\mu}_3 = \bar{\mu}$

$H_1 : \text{minimal terdapat satu } \bar{\mu}_c \neq \bar{\mu} \text{ untuk } c = 1, 2, 3$

Nilai statistik uji *wilk's lambda* yang diperoleh adalah sebesar 0,45167 dengan nilai transformasi dalam F sebesar 2,428, dan p -value sebesar 0,0624 (Lampiran 30). Dengan menggunakan taraf kepercayaan 90% atau nilai α sebesar 10%, diperoleh hasil bahwa p -value lebih kecil dari α , sehingga dapat dinyatakan bahwa tolak H_0 . Hal ini berarti bahwa nilai rata-rata dari setiap kelompok adalah berbeda secara multivariat. Oleh karena itu, dapat dinyatakan bahwa hasil pengelompokan dengan metode ensemble ROCK telah menghasilkan kelompok-kelompok yang berbeda secara signifikan.

Pengelompokan ensemble ROCK dengan pendekatan CEBMDC dapat menghasilkan kelompok-kelompok yang lebih homogen dalam kelompok dan heterogen antar kelompok dibandingkan dengan metode ensemble SWFM. Selain itu, pengelompokan ensemble ROCK memiliki kombinasi analisis pengelompokan yang lebih sedikit dibandingkan dengan pengelompokan SWFM. Meskipun memiliki tahapan analisis yang sama, metode ensemble SWFM menambahkan kombinasi analisis jarak yang baru untuk pembentukan *final cluster*. Ditinjau dari hasil pengelompokan yang dilakukan pada studi kasus, penambahan kombinasi

analisis ini mampu menyebabkan munculnya kesalahan baru dalam pengelompokan. Hal ini dibukti dengan nilai rasio metode ensembel SWFM yang lebih tinggi dibandingkan metode ensembel ROCK meskipun memiliki jumlah kelompok yang sama. Oleh karena itu dapat dinyatakan bahwa pengelompokan ensembel CEBMDC dengan pendekatan metode ROCK memberikan hasil yang lebih baik dalam pengelompokan data berskala campuran numerik dan kategorik.

BAB V

KESIMPULAN DAN SARAN

Pada bab ini dijabarkan kesimpulan dan saran yang diperoleh dari analisis dan pembahasan yang diperoleh dari BAB IV. Pada bagian awal bab ini menjabarkan beberapa kesimpulan yang diperoleh berdasarkan penjabaran algoritma dan pemograman, serta berdasarkan hasil analisis pada studi kasus yang dimiliki. Bagian akhir pada bab ini menjabarkan beberapa saran yang dapat diberikan peneliti terkait perbaikan maupun pengembangan yang lebih lanjut.

5.1. Kesimpulan

Berdasarkan hasil analisis dan pembahasan yang telah dilakukan, maka diperoleh beberapa kesimpulan antara lain,

1. Algoritma yang dibangun untuk melakukan analisis metode ensemble ROCK dan SWFM berdasarkan algoritma CEBMDC dapat digunakan untuk menangani permasalahan dalam melakukan pengelompokan data berskala campuran numerik dan kategorik. Pemograman pada *software R Project* yang dibangun dapat mempermudah dilakukannya analisis dan perbandingan dari kedua metode tersebut.
2. Pengelompokan aksesi jeruk hasil fusi protoplasma antara jeruk Siam Madu dan jeruk Satsuma Mandarin menggunakan metode ensemble ROCK dengan nilai θ sebesar 0,27 menghasilkan tiga kelompok dengan nilai ratio sebesar 0,1358, sedangkan pengelompokan dengan metode ensemble SWFM menghasilkan dua kelompok dengan nilai ratio sebesar 0,3059. Kedua analisis dilakukan dengan kombinasi pengelompokan data numerik menggunakan metode hirarki *agglomerative* dengan jarak *Euclidean* dan teknik *single linkage*, serta pengelompokan data kategorik menggunakan metode ROCK dengan nilai θ sebesar 0,17.
3. Berdasarkan dengan perbandingan nilai ratio S_W dan S_B yang terbentuk, metode ensemble ROCK memiliki nilai ratio yang lebih kecil, sehingga

dapat dinyatakan bahwa metode ensemble ROCK memberikan kinerja pengelompokan yang lebih baik dibandingkan metode ensemble SWFM. Selain menghasilkan nilai ratio yang lebih kecil, metode ensemble ROCK juga memberikan kombinasi analisis yang lebih sederhana dibandingkan metode pengelompokan ensemble SWFM.

5.2. Saran

Penelitian yang telah dilakukan ini masih terdapat beberapa perbaikan dan pengembangan selanjutnya, diantaranya:

- a. Pemograman yang dilakukan hanya sebatas menggunakan *syntax* yang dibangun pada *software R Project*, sehingga dapat dikembangkan dalam bentuk *Graphical User Interface (GUI)*.
- b. Pendekatan pengelompokan data numerik pada penelitian ini adalah dengan metode hirarki *agglomerative* dengan jarak *Euclidean* dan teknik *single linkage*, *complete linkage* dan *average linkage*, sehingga masih terdapat beberapa metode pengelompokan data numerik lain, ukuran jarak lain, dan jenis teknik pengelompokan lain yang dapat digunakan.
- c. Pendekatan pengelompokan data kategorik pada penelitian ini adalah dengan metode ROCK, sehingga dilakukan pengembangan dengan metode pengelompokan data kategorik lain seperti metode *K-Prototype* dan *Clustering Categorical Data Using Summaries (CACTUS)*.
- d. Pendekatan pengelompokan ensemble pada penelitian ini adalah dengan algoritma CEBMDC, sehingga dilakukan pengembangan dengan pendekatan lain seperti *consensus function* dan *genetic algorithm*.
- e. Penelitian ini dilakukan dengan melakukan perbandingan dua metode (ensemble ROCK dan ensemble SWFM) untuk pengelompokan data berskala campuran numerik dan kategorik, sehingga dapat dilakukan kajian perbandingan dengan metode pengelompokan data campuran lain seperti *Two Step Cluster*.

DAFTAR PUSTAKA

- Agresti, A., (2007), *Categorical Data Analysis* (second ed.), John Wiley & Sons, INC, New York.
- Anderson, T. W. dan Scluve, S. L., (1974), *Introductory Statistical Analysis*, Houghton Mifflin, Boston.
- Bunkers, M. J. dan James, R. M., (1996), "Definition of Climate Regions in the Northern Plains Using an Objective Cluster Modification Technique", *J.Climate*, hal 130-146.
- Departmen Pertanian, (2012), *Kajian Umum Mengenai Tanaman Jeruk*. 2, diakses November 22, 2015, dari <http://ditlin.hortikultura.go.id/>.
- Dewangan, R. R., Sharma, L. K., dan Akasapu, A. K., (2010), "Fuzzy Clustering Technique for Numerical and Categorical Dataset", *International Journal on Computer Science and Engineering*, hal 75-80.
- Dewi, A., (2012), *Metode Cluster Ensemble Untuk Pengelompokkan Desa Perdesaan di Provinsi Riau*, Statistika, Institut Teknologi Sepuluh Nopember, Surabaya.
- Dutta, M., Mahanta, A. K., dan Arun, K. P., (2005), "QROCK : A Quick Version of the ROCK Algorithm for Clustering of Categorical Data", *Proceedings of the 15 IEEE International Conference on Data Engineering*.
- Guha, S., Rastogi, R., dan Shim, K., (2000), "ROCK: A Robust Clustering Algorithm for Categorical Attributes", *Proceedings of the 15th International Conference on Data Engineering*.
- Hair, J. F., Black, W. C., Babin, J. B., dan Anderson, E. R., (2009), *Multivariate Data Analysis* (seventh ed.), Prentice Hall Inc, New Jersey.
- Halkidi, M., Batistakis, Y., dan Vazirgiannis, M., (2001), "On Clustering Validation Techniques". *Journal of Intelligent Information Systems*, 17:2/3, hal 107-145.
- He, Z., Xu, X., dan Deng, S., (2005a), "A Cluster Ensemble Method For Clustering Categorical Data", *Information Fusion*, 6, hal 143-151.

- He, Z., Xu, X., dan Deng, S., (2005b), "Clustering Mixed Numeric and Categorical Data: A Cluster Ensemble Approach". *Department of Computer Science and Engineering, Harbin Institute of Technology*.
- Husni, A., (2010), *Fusi Protoplas Interspecies antara Jeruk Siam Simadu (Citrus nobilis Lour.) dengan Mandarin Satsuma (C. unshiu Marc.)*, Agronomi, Institut Pertanian Bogor, Bogor.
- Irfandi, Y. P., (2016), *Penerapan Algoritma Squeezer untuk Pengelompokan Tanaman Jeruk Hasil Fusi Berdasarkan Karakteristik Kualitatif dan Kuantitatif*, Statistika, Institut Teknologi Sepuluh Nopember, Surabaya.
- Johnson, R. A., dan Winchern, D. W., (2007), *Applied Multivariate Statistical Analysis* (sixth ed.), Pearson Education, Inc, New Jersey.
- Kader, D. G. dan Perry, M., (2007), "Variability for Categorical Variables", *Journal of Statistics Education*, Vol 15, No. 2.
- Mardia, K. V., (1970), "Measures of Multivariate Skewness and Kurtosis with Applications". *Biometrika*. Vol. 50.
- Martasari, C., (2014), *Kajian Genetik dan Percepatan Pembungaan Tanaman Hasil Fusi Protoplasma Jeruk Siam Madu dan Satsuma Mandarin*, Universitas Brawijaya, Malang.
- Mulyono, M. S., (2006), *Pengelompokan Data Kategorik dengan Algoritma ROCK*, Universitas Airlangga, Surabaya.
- Light, R. J., dan Margolin, B. H., (1971), "An Analysis of Variance for Categorical Data", *Journal of American Statistical Association*, Vol. 66, No.335.
- Okada, T., (1999), "Sum of Square Decomposition for Categorical Data", *Kwansei Gakuin Studies in Computer Science*, Vol 14.
- Rahayu, D. P., (2013), "Analisis Karakteristik Kelompok dengan Menggunakan Cluster Ensemble", *Jurnal Matematika, Sains, dan Teknologi*, Vol 14, No 1.
- Reddy, M. V. J. dan Kavitha, B., (2010), "Efficient Ensemble Algorithm for Mixed Numeric and Categorical Data", *Computational Intelligence and Computing research (ICCIC)*, IEEE International Conference.

- Reddy, M. V. J. dan Kavitha, B., (2012), "Clustering the Mixed Numerical and Categorical Dataset using Similarity Weight and Filter Method". *International Journal of Database Theory and Application*, Vol 5, No 1.
- Saputra, C. W., (2016), *Pengelompokan Akses Jeruk Persilangan Berdasarkan Karakteristik Kuantitatif dan Kualitatif Menggunakan Fuzzy C-Means dan K-Modes*, Statistika, Institut Teknologi Sepuluh Nopember, Surabaya.
- Sharma, S., (1996), *Applied Multivariate Technique*, John Wiley and Sons, Inc, New York.
- Suguna, J., dan Selvi, M. A., (2012), "Ensemble Fuzzy Clustering for Mixed Numerical and Categorical Data", *International Journal of Computer Application*, 42, 19.
- Tyagi, A. dan Sharma, S., (2012), "Implementation of ROCK Clustering Algorithm for the Optimazation of Query Searching Time", *International Journal on Computer Science and Engineering*, Vol 4, No 05.
- Yoon, H. S., Ahn, S. Y., Lee, S. H., Cho, S. B., dan Kim, J. H., (2006), "Heterogeneous Clustering Ensemble Method For Combining Different Cluster Results". *BioDM 2006*, hal 82-91.

“Halaman ini sengaja dikosongkan”

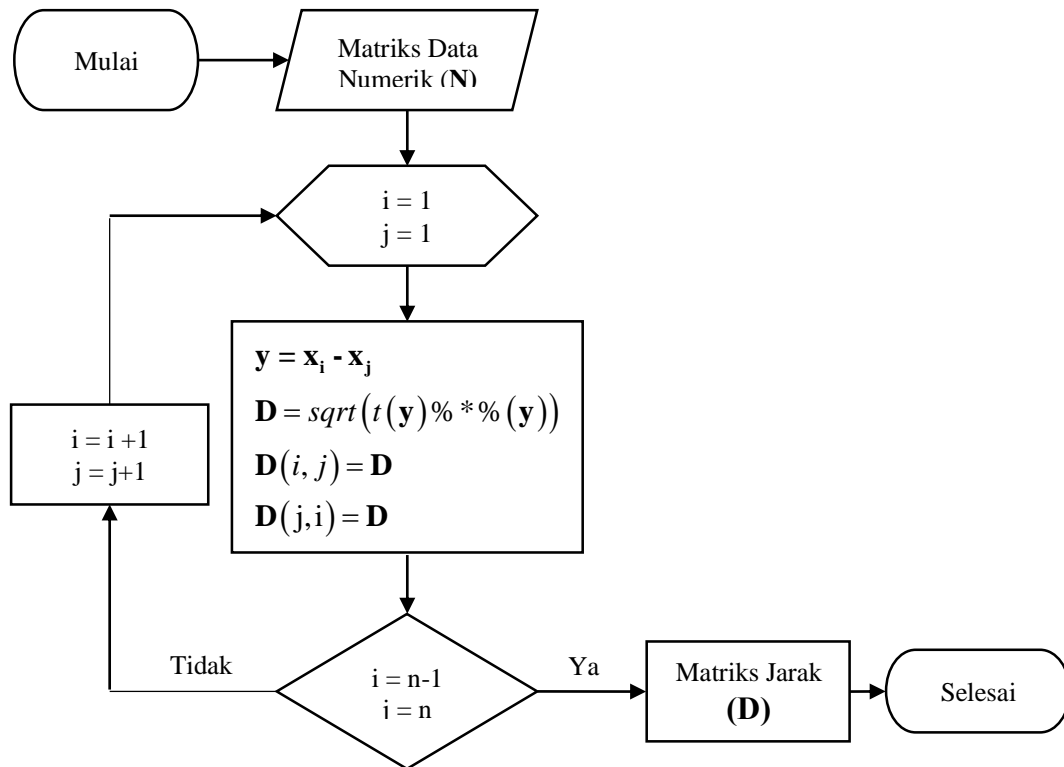
Lampiran 1.Data Pengamatan Variabel Numerik

tid	No. Akses	Diameter Buah	Tebal Kulit	Jumlah Juring	Tebal daging Kulit	Diameter Axis	Jumlah Biji	Berat Buah	Brix
No	Nama	X1	X2	X3	X4	X5	X6	X7	X8
1	FS01	62,32	2,40	10	1,78	13,65	21,10	94,44	10,20
2	FS07	51,23	1,99	10	1,86	10,58	20,90	67,07	12,80
3	FS10	49,95	1,26	10	1,88	8,81	19,30	92,99	14,00
4	FS14	51,77	2,01	9	2,32	12,78	18,25	161,45	14,40
5	FS15	69,91	2,11	10	2,62	18,53	18,10	107,22	10,00
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
21	FS103	48,55	1,56	10	1,43	10,97	13,00	108,66	13,60
22	FS107	60,69	1,98	11	1,92	13,23	15,80	72,69	11,00
23	P III i	60,32	2,00	9	1,61	9,46	4,00	108,92	11,00
24	TL 3	50,10	1,31	9	1,60	12,12	4,00	65,18	14,00
25	TL 5	50,11	1,98	10	1,95	11,60	4,00	67,76	12,20

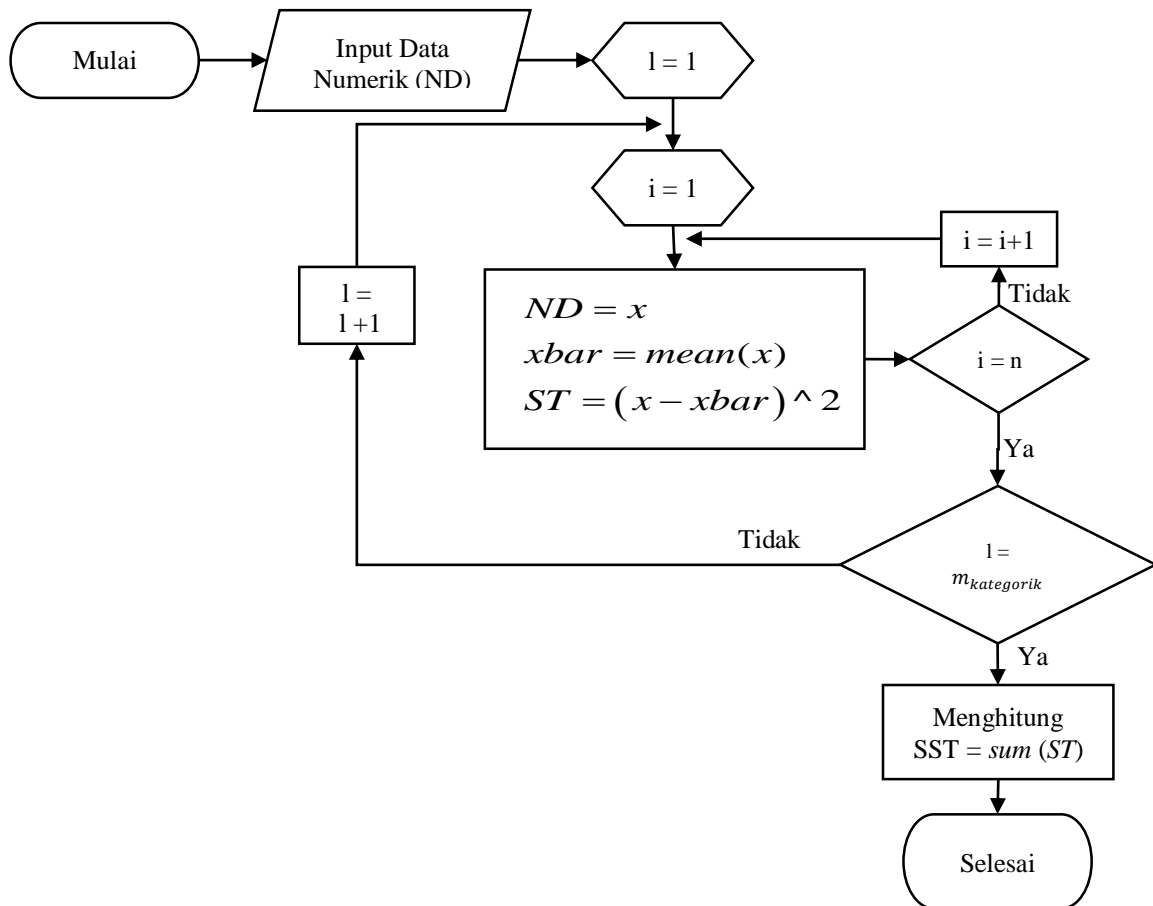
Lampiran 2. Data Pengamatan Variabel Kategorik

tid	No. Akses	Bentuk Buah	Warna Kulit	Permukaan Kulit Buah	Axis Buah	Warna Daging Kulit	Tampak Melintang axis	Tekstur Pulp
No	Nama	X9	X10	X11	X12	X13	X14	X15
1	FS01	1	4	1	3	1	1	2
2	FS07	5	4	1	3	1	1	2
3	FS10	1	4	1	3	1	2	2
4	FS14	1	4	1	3	1	1	2
5	FS15	1	5	2	3	1	2	3
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
21	FS103	1	4	1	2	1	1	1
22	FS107	5	4	1	3	2	1	1
23	P III i	1	4	1	3	1	1	1
24	TL 3	1	4	1	2	1	2	1
25	TL 5	1	4	1	3	1	1	1

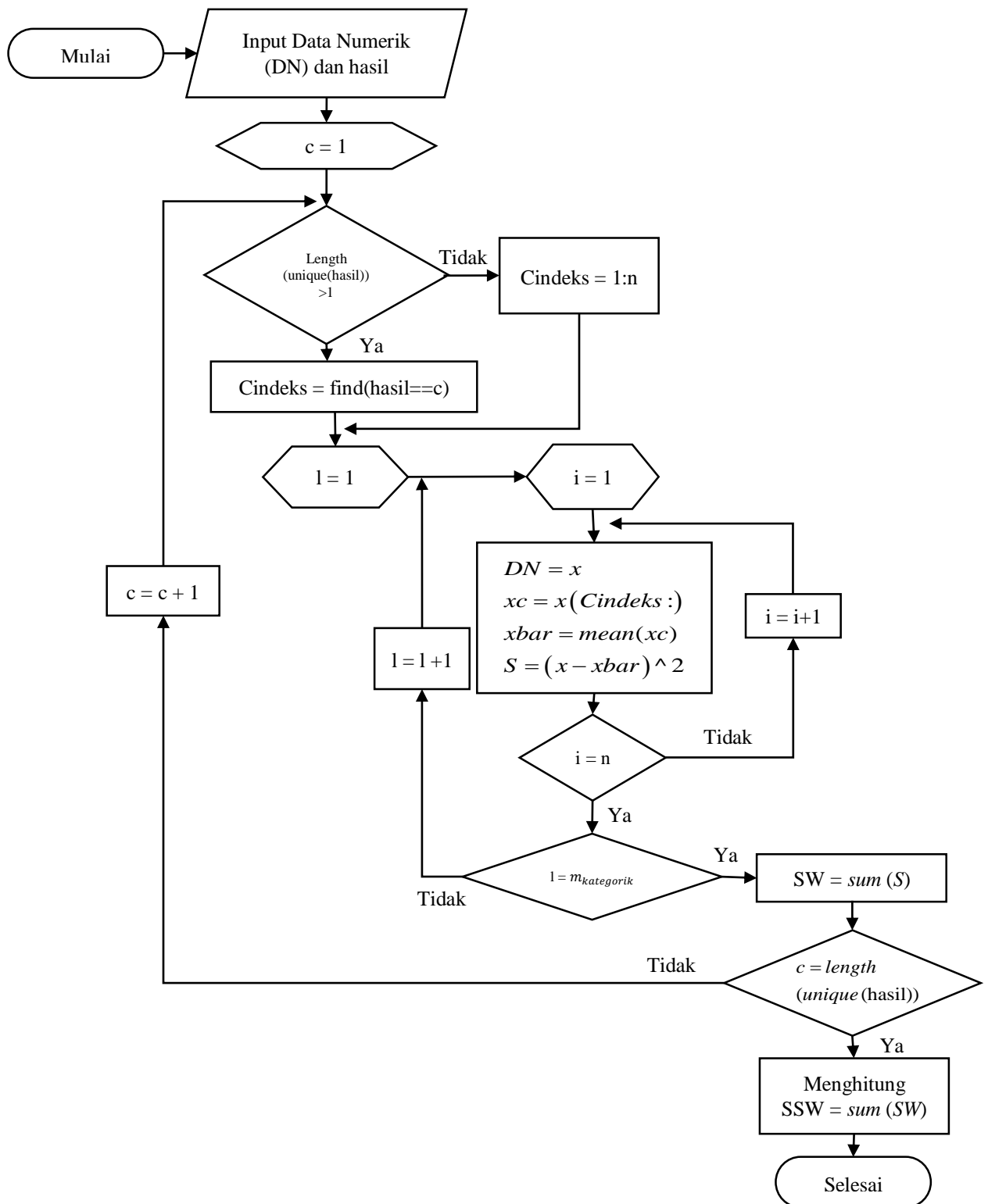
Lampiran 3. Flowchart Prosedur Perhitungan Jarak Euclidean



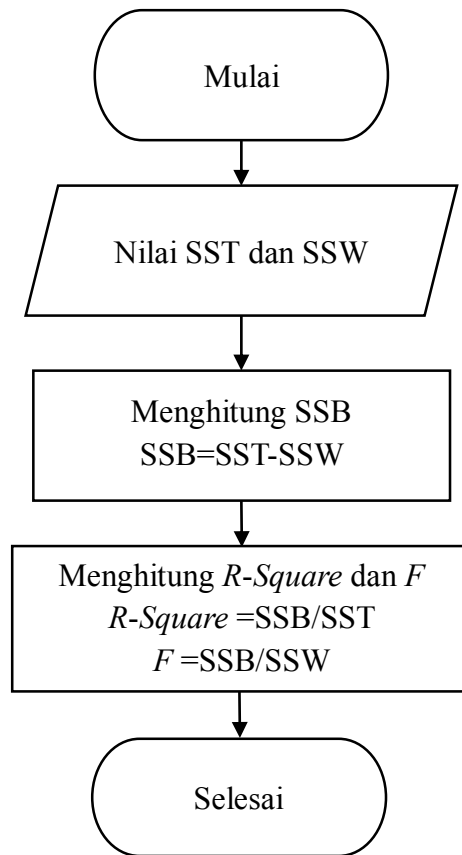
Lampiran 4. Flowchart Prosedur Perhitungan Sum Square Total (SST)



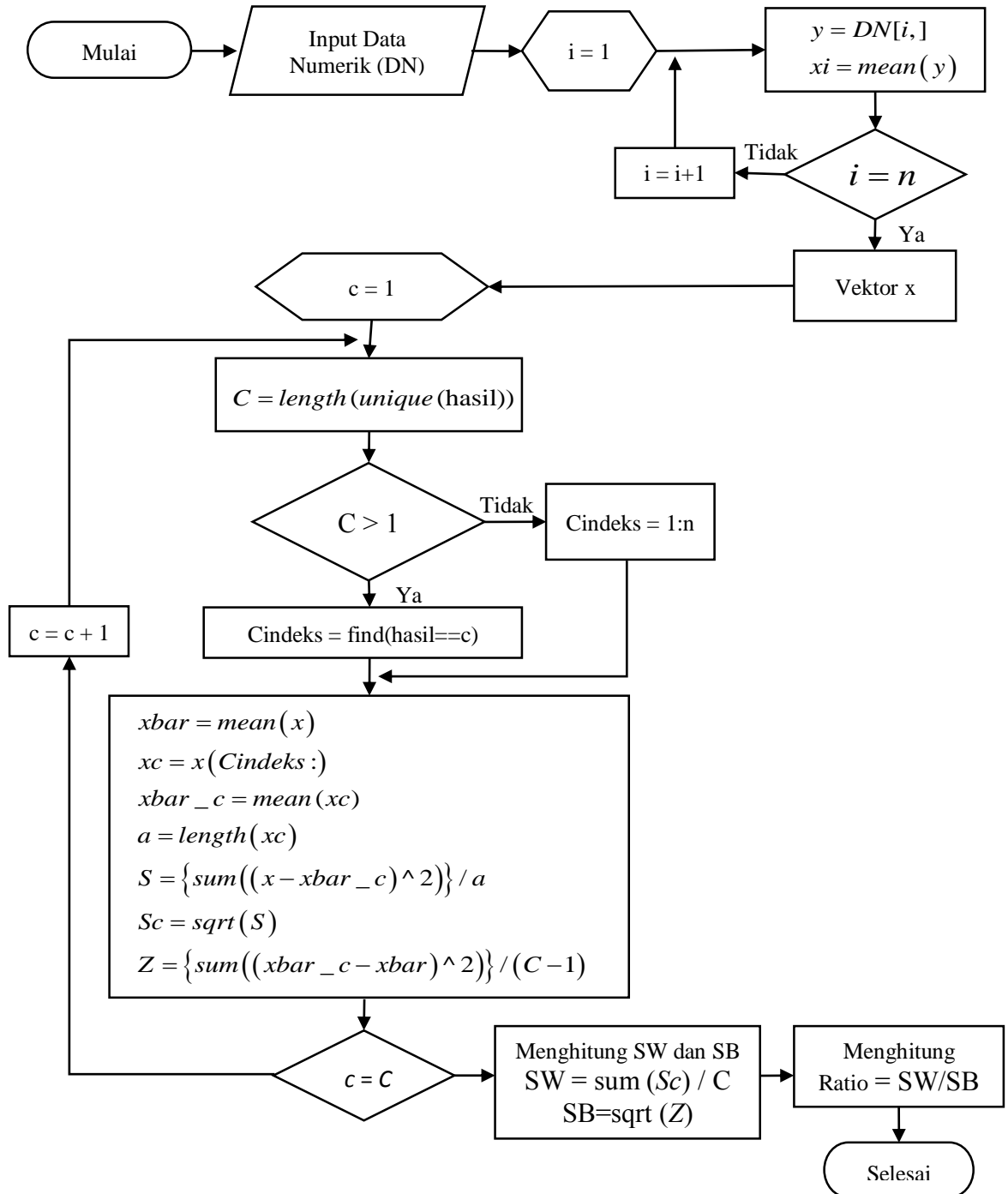
Lampiran 5. Flowchart Prosedur Perhitungan Sum Square Within (SSW)



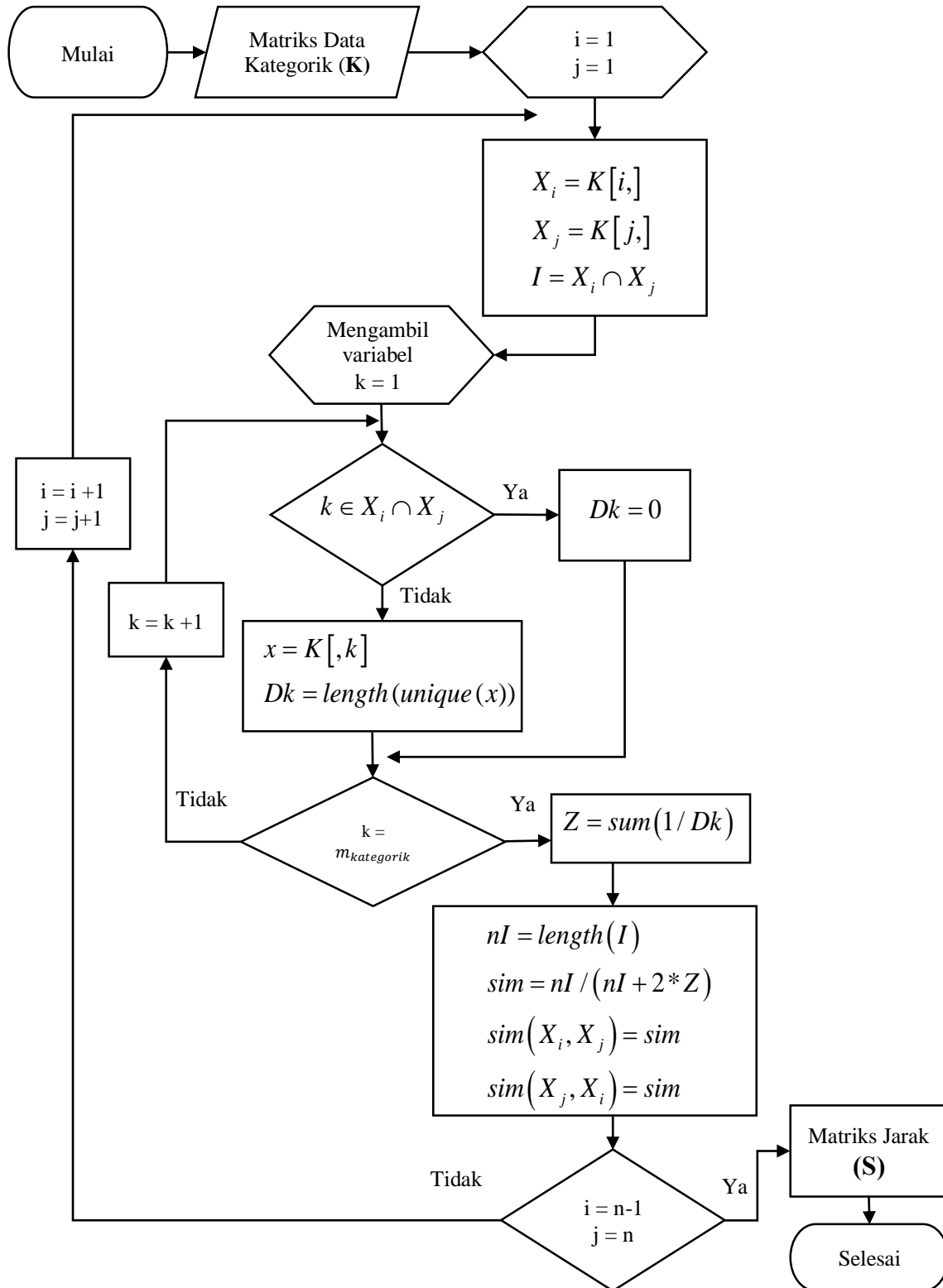
Lampiran 6. *Flowchart* Prosedur Perhitungan *R-Square* dan *F*



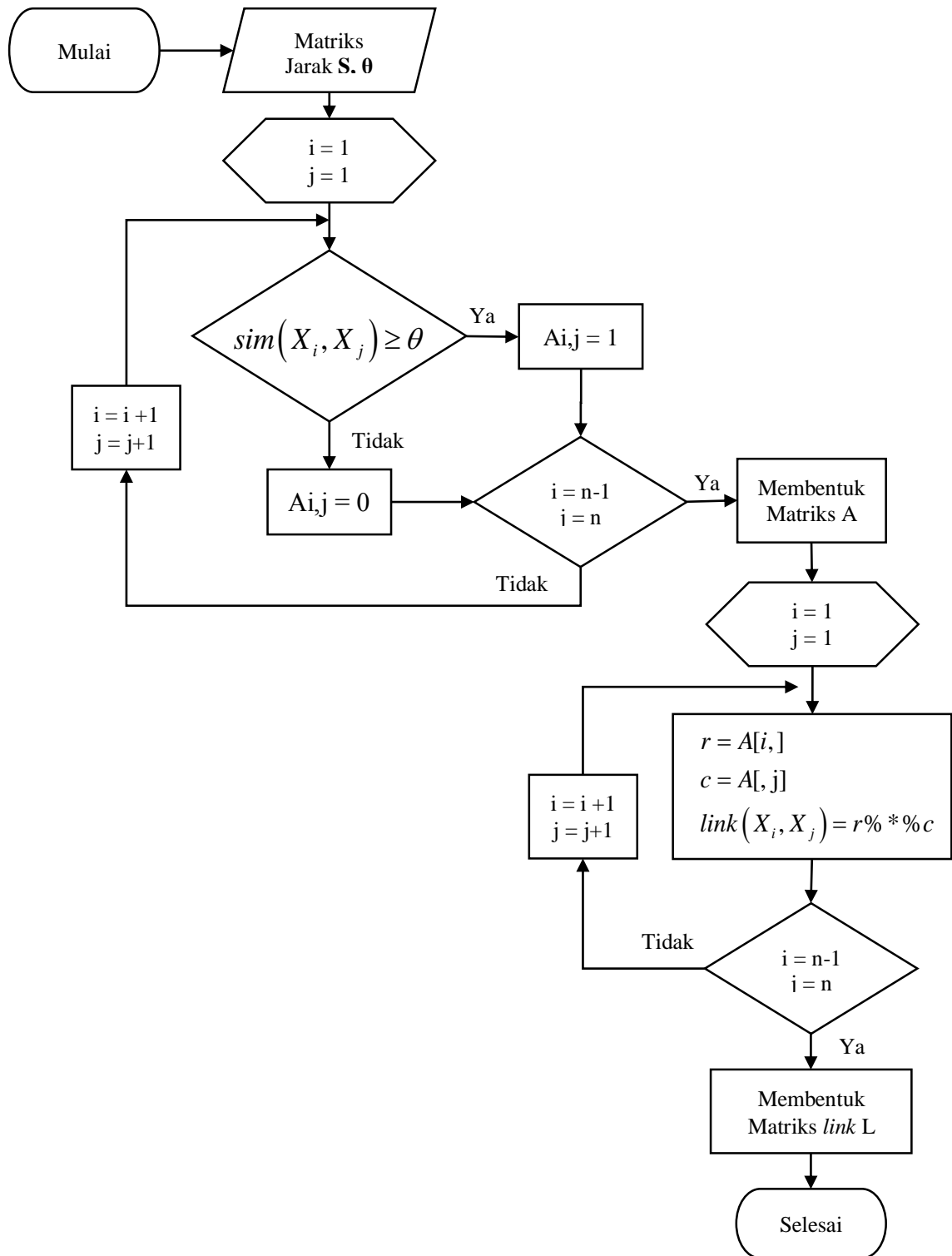
Lampiran 7. Flowchart Prosedur Perhitungan Rasio S_W dan S_B untuk Data Numerik



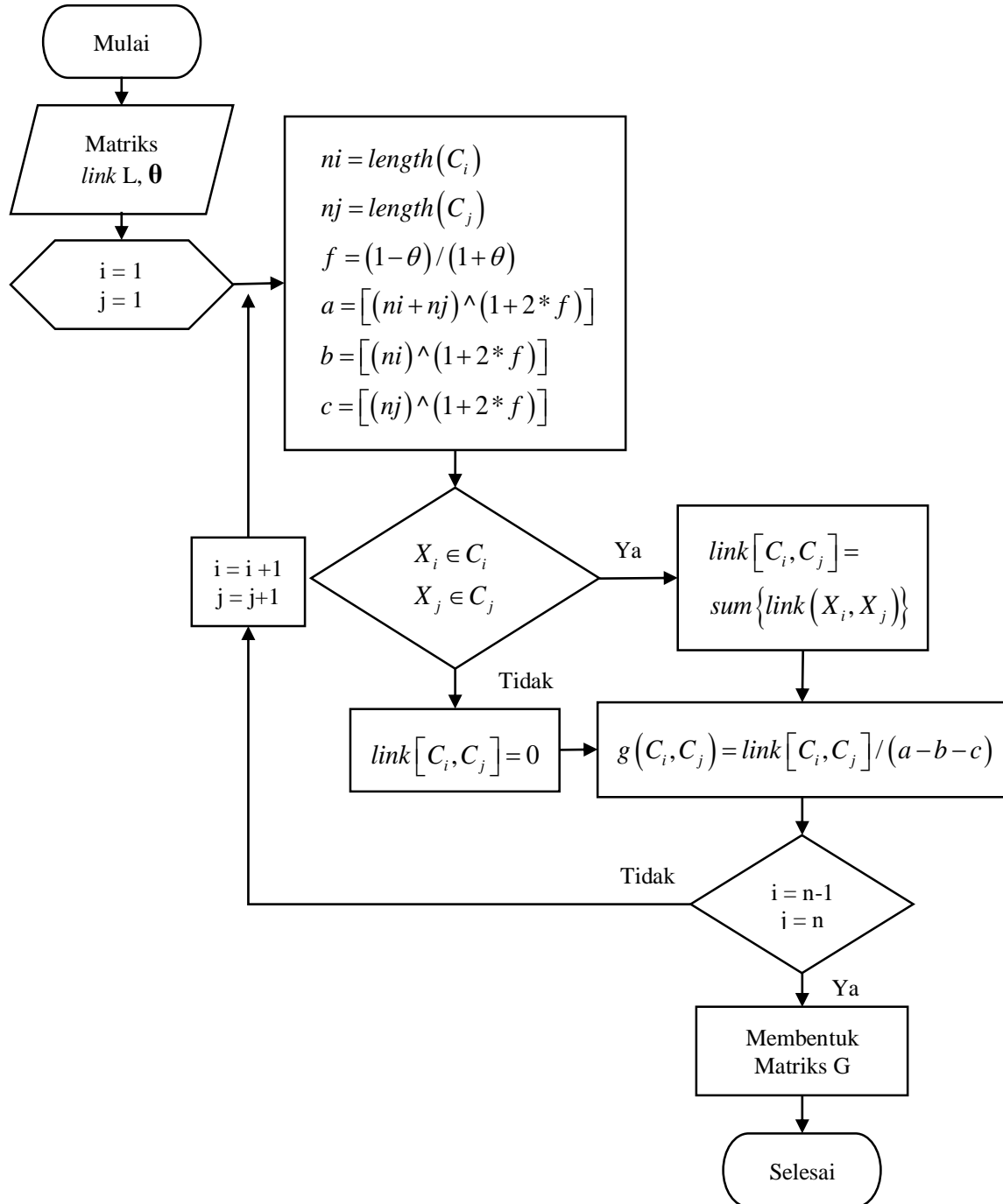
Lampiran 8. Flowchart Prosedur Perhitungan *Similarity* pada Metode ROCK



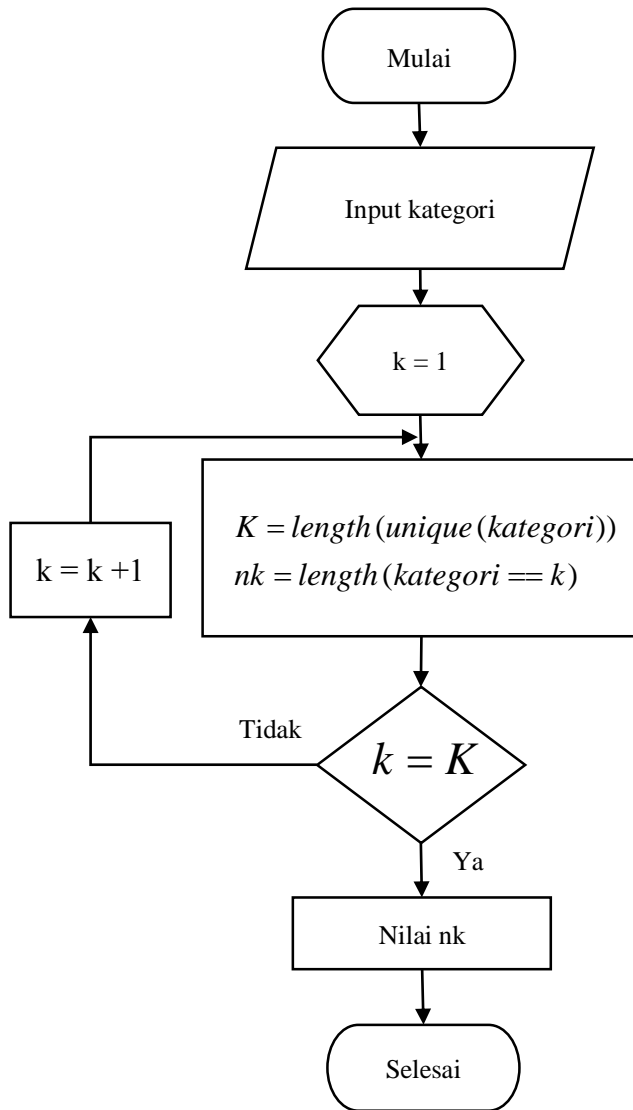
Lampiran 9. Flowchart Prosedur Perhitungan *Link* pada Metode ROCK



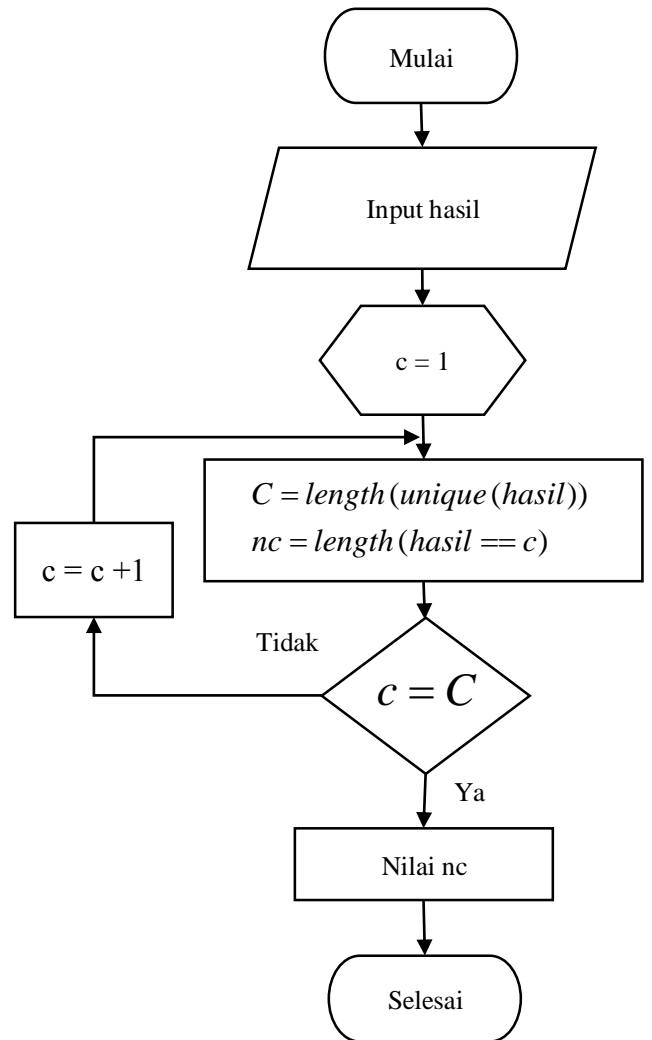
Lampiran 10. Flowchart Prosedur Perhitungan Goodness Measure pada Metode ROCK



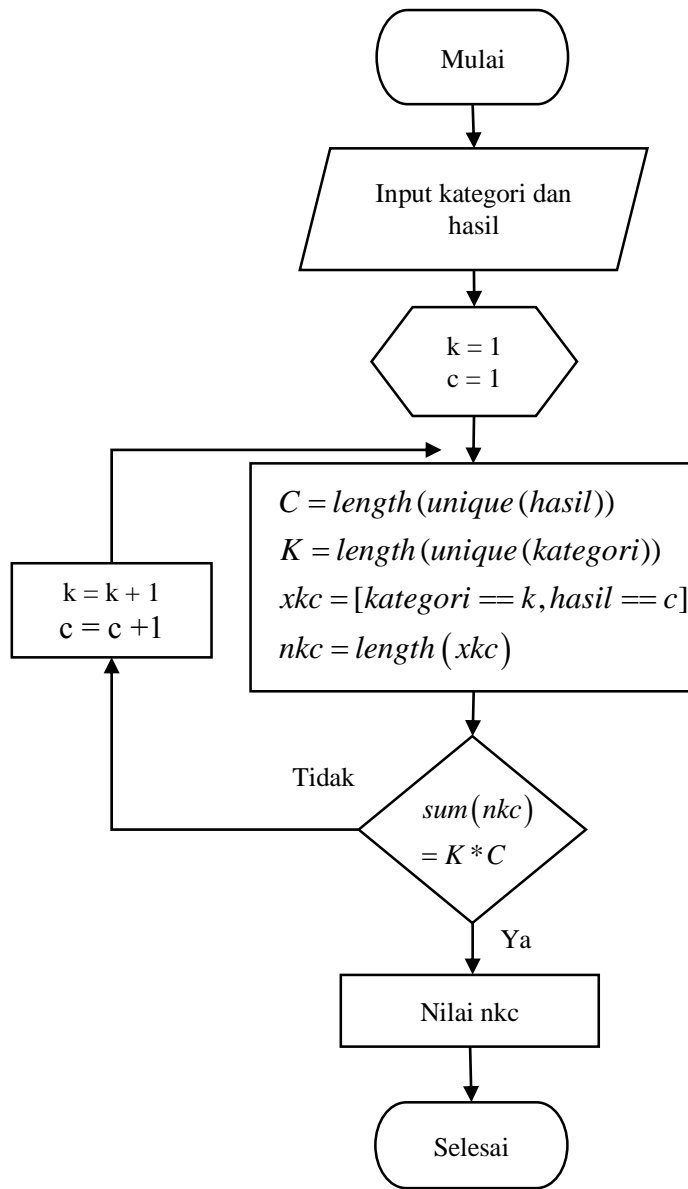
**Lampiran 11. Flowchart Prosedur Perhitungan (a) Nilai n_k , (b) Nilai n_c ,
(c) Nilai n_{kc} , dan (d) Rasio S_W dan S_B untuk Data Kategorik**



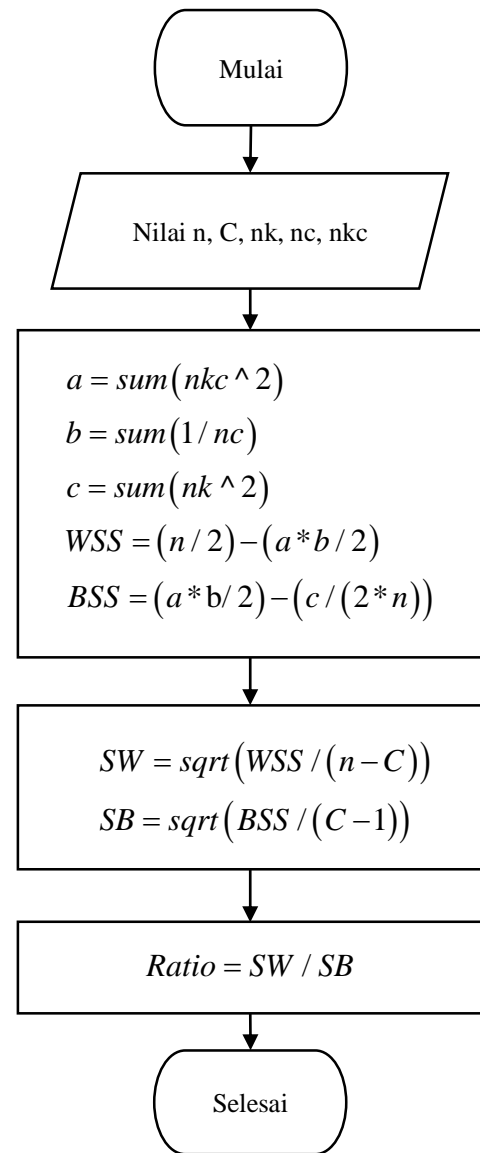
(a)



(b)

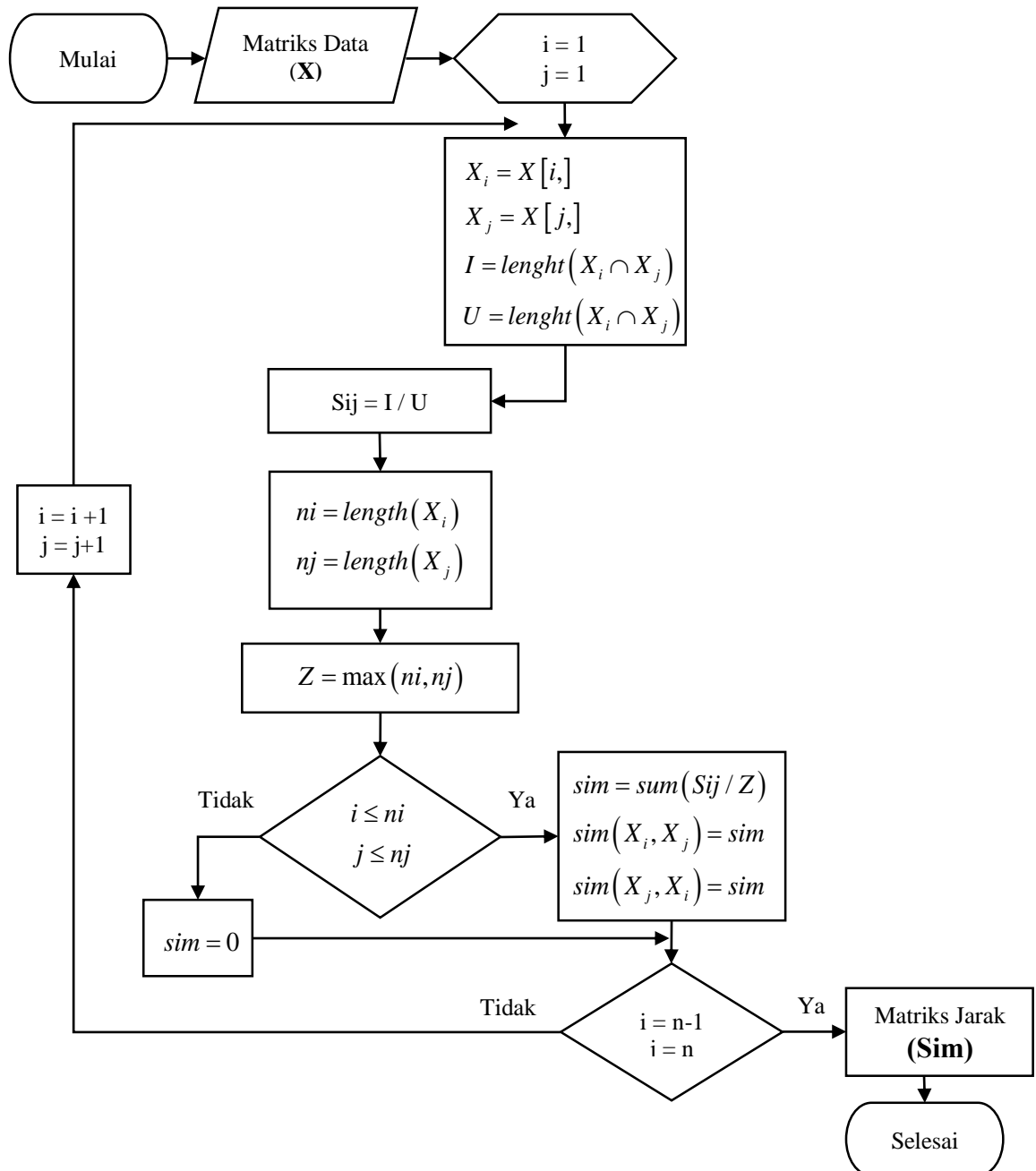


(c)

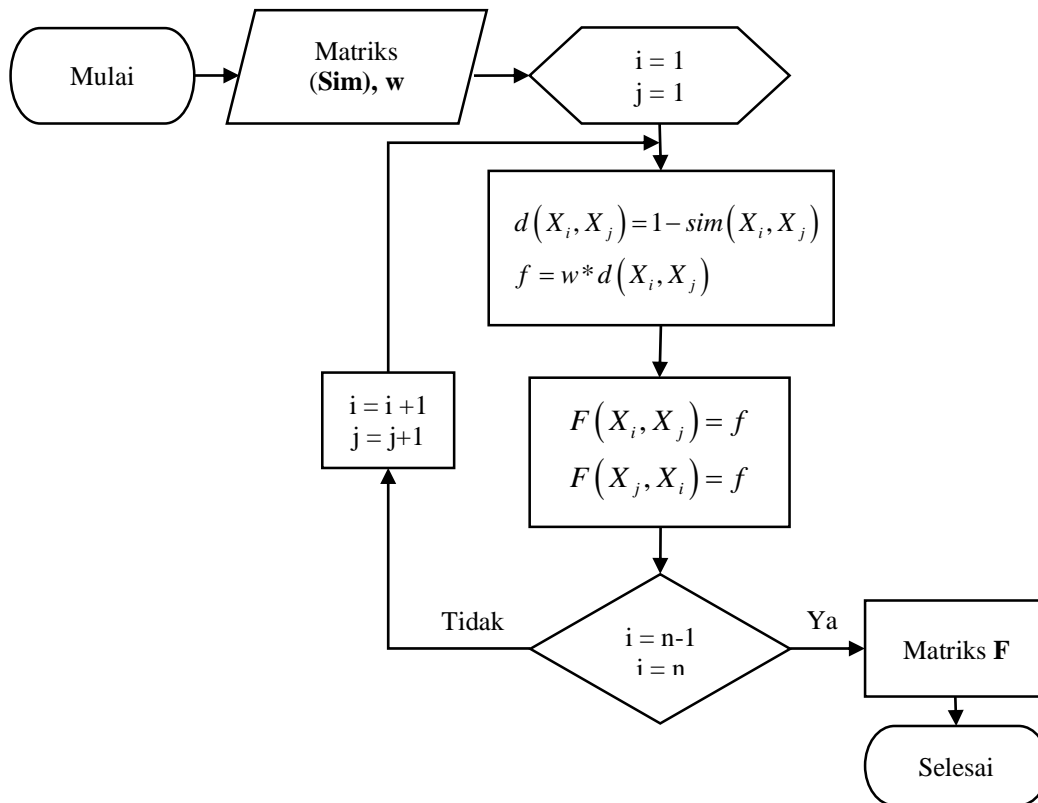


(d)

Lampiran 12. Flowchart Prosedur Perhitungan *Similarity Weight*



Lampiran 13. Flowchart Prosedur Perhitungan Nilai Jarak F



Lampiran 14. *Syntax* Analisis Kelompok Hirarki Agglomerative untuk Data Numerik

a. Jumlah kelompok yang dibentuk sama dengan 2

```

Analisis.kluster.k2 = function(data){
  k = 2
  datanumerik = data.frame(data$X1, data$X2, data$X3, data$X4,
  data$X5, data$X6, data$X7, data$X8)
  d = dist(datanumerik, method = "euclidean")
  # Analisis Kluster Hirarki
  fit.sin = hclust(d, method = "single")
  fit.com = hclust(d, method = "complete")
  fit.ave = hclust(d, method = "average")
  # Memotong Dendrogram untuk k Kluster
  single = cutree(fit.sin, k=k)
  complete = cutree(fit.com, k=k)
  average = cutree(fit.ave, k=k)
  hasil.kelompok.numerik = data.frame>Nama, single, complete,
  average)
  # Menghitung sum of square total  $y = (x_{il} - \bar{x}_l)^2$ 
  y = c((data$X1-mean(data$X1))^2, (data$X2-mean(data$X2))^2,
  (data$X3-mean(data$X3))^2, (data$X4-mean(data$X4))^2,
  (data$X5-mean(data$X5))^2, (data$X6-mean(data$X6))^2,
  (data$X7-mean(data$X7))^2, (data$X8-mean(data$X8))^2)
  SST = sum (y)
  # Menghitung sum of square within group (single)
  kelompok.single = hasil.kelompok.numerik$single
  data.single = data.frame(single, data$X1, data$X2,
  data$X3, data$X4, data$X5, data$X6, data$X7, data$X8)
  data.single.sort = data.single[order(data.single$single),]
  xlc1bar.single = matrix(c(rep(mean(data.c1.single$data.X1),a),
  rep(mean(data.c1.single$data.X2),a),
  rep(mean(data.c1.single$data.X3),a),
  rep(mean(data.c1.single$data.X4),a),
  rep(mean(data.c1.single$data.X5),a),
  rep(mean(data.c1.single$data.X6),a),
  rep(mean(data.c1.single$data.X7),a),
  rep(mean(data.c1.single$data.X8),a)), ncol = 8)
  xlc2bar.single = matrix(c(rep(mean(data.c2.single$data.X1),b),
  rep(mean(data.c2.single$data.X2),b),
  rep(mean(data.c2.single$data.X3),b),
  rep(mean(data.c2.single$data.X4),b),
  rep(mean(data.c2.single$data.X5),b),
  rep(mean(data.c2.single$data.X6),b),
  rep(mean(data.c2.single$data.X7),b),
  rep(mean(data.c2.single$data.X8),b)), ncol = 8)
  xil.c1.single =
  matrix(c(data.c1.single$data.X1,data.c1.single$data.X2,
  data.c1.single$data.X3,data.c1.single$data.X4,
  data.c1.single$data.X5,data.c1.single$data.X6,
  data.c1.single$data.X7,data.c1.single$data.X8), ncol = 8)
  xil.c2.single =
  matrix(c(data.c2.single$data.X1,data.c2.single$data.X2,
  data.c2.single$data.X3,data.c2.single$data.X4,
  data.c2.single$data.X5,data.c2.single$data.X6,

```

```

data.c2.single$data.X7,data.c2.single$data.X8), ncol = 8)
x1.single = sum ((xil.c1.single-xlc1bar.single)^2)
x2.single = sum ((xil.c2.single-xlc2bar.single)^2)
SSW.single.c2 = sum (x1.single,x2.single)
SSB.single.c2 = SST - SSW.single.c2
R.square.single.c2 = SSB.single.c2/SST
Hasil.single = c(SST, SSW.single.c2, R.square.single.c2)
# Menghitung sum of square within group (complete)
kelompok.complete = hasil.kelompok.numerik$complete
data.complete = data.frame(complete, data$X1, data$X2,
data$X3, data$X4, data$X5, data$X6, data$X7, data$X8)
data.complete.sort =
data.complete[order(data.complete$complete),]
xlc1bar.complete =
matrix(c(rep(mean(data.c1.complete$data.X1),a),
rep(mean(data.c1.complete$data.X2),a),
rep(mean(data.c1.complete$data.X3),a),
rep(mean(data.c1.complete$data.X4),a),
rep(mean(data.c1.complete$data.X5),a),
rep(mean(data.c1.complete$data.X6),a),
rep(mean(data.c1.complete$data.X7),a),
rep(mean(data.c1.complete$data.X8),a)), ncol = 8)
xlc2bar.complete =
matrix(c(rep(mean(data.c2.complete$data.X1),b),
rep(mean(data.c2.complete$data.X2),b),
rep(mean(data.c2.complete$data.X3),b),
rep(mean(data.c2.complete$data.X4),b),
rep(mean(data.c2.complete$data.X5),b),
rep(mean(data.c2.complete$data.X6),b),
rep(mean(data.c2.complete$data.X7),b),
rep(mean(data.c2.complete$data.X8),b))), ncol = 8)
xil.c1.complete =
matrix(c(data.c1.complete$data.X1,data.c1.complete$data.X2,
data.c1.complete$data.X3,data.c1.complete$data.X4,
data.c1.complete$data.X5,data.c1.complete$data.X6,
data.c1.complete$data.X7,data.c1.complete$data.X8), ncol = 8)
xil.c2.complete =
matrix(c(data.c2.complete$data.X1,data.c2.complete$data.X2,
data.c2.complete$data.X3,data.c2.complete$data.X4,
data.c2.complete$data.X5,data.c2.complete$data.X6,
data.c2.complete$data.X7,data.c2.complete$data.X8), ncol = 8)
x1.complete = sum ((xil.c1.complete-xlc1bar.complete)^2)
x2.complete = sum ((xil.c2.complete-xlc2bar.complete)^2)
SSW.complete.c2 = sum (x1.complete,x2.complete)
SSB.complete.c2 = SST - SSW.complete.c2
R.square.complete.c2 = SSB.complete.c2/SST
Hasil.complete = c(SST, SSW.complete.c2, R.square.complete.c2)
# Menghitung sum of square within group (average)
kelompok.average = hasil.kelompok.numerik$average
data.average = data.frame(average, data$X1, data$X2,
data$X3, data$X4, data$X5, data$X6, data$X7, data$X8)
data.average.sort = data.average[order(data.average$average),]
xlc1bar.average = matrix(c(rep(mean(data.c1.average$data.X1),a),
rep(mean(data.c1.average$data.X2),a),
rep(mean(data.c1.average$data.X3),a),
rep(mean(data.c1.average$data.X4),a),
rep(mean(data.c1.average$data.X5),a),

```

```

rep(mean(data.c1.average$data.X6),a),
rep(mean(data.c1.average$data.X7),a),
rep(mean(data.c1.average$data.X8),a)), ncol = 8)
xlc2bar.average = matrix(c(rep(mean(data.c2.average$data.X1),b),
rep(mean(data.c2.average$data.X2),b),
rep(mean(data.c2.average$data.X3),b),
rep(mean(data.c2.average$data.X4),b),
rep(mean(data.c2.average$data.X5),b),
rep(mean(data.c2.average$data.X6),b),
rep(mean(data.c2.average$data.X7),b),
rep(mean(data.c2.average$data.X8),b))), ncol = 8)
xil.c1.average =
matrix(c(data.c1.average$data.X1,data.c1.average$data.X2,
data.c1.average$data.X3,data.c1.average$data.X4,
data.c1.average$data.X5,data.c1.average$data.X6,
data.c1.average$data.X7,data.c1.average$data.X8), ncol = 8)
xil.c2.average =
matrix(c(data.c2.average$data.X1,data.c2.average$data.X2,
data.c2.average$data.X3,data.c2.average$data.X4,
data.c2.average$data.X5,data.c2.average$data.X6,
data.c2.average$data.X7,data.c2.average$data.X8), ncol = 8)
x1.average = sum ((xil.c1.average-xlc1bar.average)^2)
x2.average = sum ((xil.c2.average-xlc2bar.average)^2)
SSW.average.c2 = sum (x1.average,x2.average)
SSB.average.c2 = SST - SSW.average.c2
R.square.average.c2 = SSB.average.c2/SST
Hasil.average = c(SST, SSW.average.c2, R.square.average.c2)
Hasil = rbind(Hasil.single, Hasil.complete, Hasil.average)
print("Hasil Pengelompokan Hirarki")
print(hasil.kelompok.numerik)
print("-----")
print(" Untuk 2 kelompok yang terbentuk ")
print("Metode          SST          SSW          R-square")
print(Hasil)
}

```

b. Jumlah kelompok yang dibentuk sama dengan 3

```

Analisis.kluster.k3 = function(data){
k
= 3
datanumerik = data.frame(data$X1, data$X2, data$X3, data$X4,
data$X5, data$X6, data$X7, data$X8)
d
= dist(datanumerik, method = "euclidean")
# Analisis Kluster Hirarki
fit.sin
= hclust(d, method = "single")
fit.com
= hclust(d, method = "complete")
fit.ave
= hclust(d, method = "average")
# Memotong Dendrogram untuk k Kluster
single
= cutree(fit.sin, k=k)
complete
= cutree(fit.com, k=k)
average
= cutree(fit.ave, k=k)
hasil.kelompok.numerik = data.frame>Nama, single, complete,
average)
# Menghitung sum of square total y = (xil-xlbar)^2
y = c((data$X1-mean(data$X1))^2, (data$X2-mean(data$X2))^2,
(data$X3-mean(data$X3))^2, (data$X4-mean(data$X4))^2,

```

```

(data$X5-mean(data$X5))^2, (data$X6-mean(data$X6))^2,
(data$X7-mean(data$X7))^2, (data$X8-mean(data$X8))^2)
SST = sum (y)
# Menghitung sum of square within group (single)
kelompok.single = hasil.kelompok.numerik$single
data.single = data.frame(single, data$X1, data$X2,
data$X3, data$X4, data$X5, data$X6, data$X7, data$X8)
data.single.sort = data.single[order(data.single$single),]
xlc1bar.single = matrix(c(rep(mean(data.c1.single$data.X1),a),
rep(mean(data.c1.single$data.X2),a),
rep(mean(data.c1.single$data.X3),a),
rep(mean(data.c1.single$data.X4),a),
rep(mean(data.c1.single$data.X5),a),
rep(mean(data.c1.single$data.X6),a),
rep(mean(data.c1.single$data.X7),a),
rep(mean(data.c1.single$data.X8),a)), ncol = 8)
xlc2bar.single = matrix(c(rep(mean(data.c2.single$data.X1),b),
rep(mean(data.c2.single$data.X2),b),
rep(mean(data.c2.single$data.X3),b),
rep(mean(data.c2.single$data.X4),b),
rep(mean(data.c2.single$data.X5),b),
rep(mean(data.c2.single$data.X6),b),
rep(mean(data.c2.single$data.X7),b),
rep(mean(data.c2.single$data.X8),b)), ncol = 8)
xlc3bar.single = matrix(c(rep(mean(data.c3.single$data.X1),c),
rep(mean(data.c3.single$data.X2),c),
rep(mean(data.c3.single$data.X3),c),
rep(mean(data.c3.single$data.X4),c),
rep(mean(data.c3.single$data.X5),c),
rep(mean(data.c3.single$data.X6),c),
rep(mean(data.c3.single$data.X7),c),
rep(mean(data.c3.single$data.X8),c)), ncol = 8)
xil.c1.single =
matrix(c(data.c1.single$data.X1,data.c1.single$data.X2,
data.c1.single$data.X3,data.c1.single$data.X4,
data.c1.single$data.X5,data.c1.single$data.X6,
data.c1.single$data.X7,data.c1.single$data.X8), ncol = 8)
xil.c2.single =
matrix(c(data.c2.single$data.X1,data.c2.single$data.X2,
data.c2.single$data.X3,data.c2.single$data.X4,
data.c2.single$data.X5,data.c2.single$data.X6,
data.c2.single$data.X7,data.c2.single$data.X8), ncol = 8)
xil.c3.single =
matrix(c(data.c3.single$data.X1,data.c3.single$data.X2,
data.c3.single$data.X3,data.c3.single$data.X4,
data.c3.single$data.X5,data.c3.single$data.X6,
data.c3.single$data.X7,data.c3.single$data.X8), ncol = 8)
x1.single = sum ((xil.c1.single-xlc1bar.single)^2)
x2.single = sum ((xil.c2.single-xlc2bar.single)^2)
x3.single = sum ((xil.c3.single-xlc3bar.single)^2)
SSW.single.c1 = sum (x1.single)
SSW.single.c2 = sum (x1.single,x2.single)
SSW.single.c3 = sum (x1.single,x2.single,x3.single)
SSB.single.c1 = SST - SSW.single.c1
SSB.single.c2 = SST - SSW.single.c2
SSB.single.c3 = SST - SSW.single.c3
R.square.single.c1 = SSB.single.c1/SST

```

```

R.square.single.c2 = SSB.single.c2/SST
R.square.single.c3 = SSB.single.c3/SST
Hasil.single = c(SST, SSW.single.c3,R.square.single.c3)
# Menghitung sum of square within group (complete)
kelompok.complete      = hasil.kelompok.numerik$complete
data.complete           = data.frame(complete, data$X1, data$X2,
data$X3, data$X4, data$X5, data$X6, data$X7, data$X8)
data.complete.sort      =
data.complete[order(data.complete$complete),]
data.c1.complete        = data.complete.sort[1:a,]
data.c2.complete        = data.complete.sort[(a+1):(a+b),]
data.c3.complete        = data.complete.sort[(a+b+1):(a+b+c),]
xlc1bar.complete =
matrix(c(rep(mean(data.c1.complete$data.X1),a),
rep(mean(data.c1.complete$data.X2),a),
rep(mean(data.c1.complete$data.X3),a),
rep(mean(data.c1.complete$data.X4),a),
rep(mean(data.c1.complete$data.X5),a),
rep(mean(data.c1.complete$data.X6),a),
rep(mean(data.c1.complete$data.X7),a),
rep(mean(data.c1.complete$data.X8),a)), ncol = 8)
xlc2bar.complete =
matrix(c(rep(mean(data.c2.complete$data.X1),b),
rep(mean(data.c2.complete$data.X2),b),
rep(mean(data.c2.complete$data.X3),b),
rep(mean(data.c2.complete$data.X4),b),
rep(mean(data.c2.complete$data.X5),b),
rep(mean(data.c2.complete$data.X6),b),
rep(mean(data.c2.complete$data.X7),b),
rep(mean(data.c2.complete$data.X8),b)), ncol = 8)
xlc3bar.complete =
matrix(c(rep(mean(data.c3.complete$data.X1),c),
rep(mean(data.c3.complete$data.X2),c),
rep(mean(data.c3.complete$data.X3),c),
rep(mean(data.c3.complete$data.X4),c),
rep(mean(data.c3.complete$data.X5),c),
rep(mean(data.c3.complete$data.X6),c),
rep(mean(data.c3.complete$data.X7),c),
rep(mean(data.c3.complete$data.X8),c)), ncol = 8)
xil.c1.complete =
matrix(c(data.c1.complete$data.X1,data.c1.complete$data.X2,
data.c1.complete$data.X3,data.c1.complete$data.X4,
data.c1.complete$data.X5,data.c1.complete$data.X6,
data.c1.complete$data.X7,data.c1.complete$data.X8), ncol = 8)
xil.c2.complete =
matrix(c(data.c2.complete$data.X1,data.c2.complete$data.X2,
data.c2.complete$data.X3,data.c2.complete$data.X4,
data.c2.complete$data.X5,data.c2.complete$data.X6,
data.c2.complete$data.X7,data.c2.complete$data.X8), ncol = 8)
xil.c3.complete =
matrix(c(data.c3.complete$data.X1,data.c3.complete$data.X2,
data.c3.complete$data.X3,data.c3.complete$data.X4,
data.c3.complete$data.X5,data.c3.complete$data.X6,
data.c3.complete$data.X7,data.c3.complete$data.X8), ncol = 8)
x1.complete = sum ((xil.c1.complete-xlc1bar.complete)^2)
x2.complete = sum ((xil.c2.complete-xlc2bar.complete)^2)
x3.complete = sum ((xil.c3.complete-xlc3bar.complete)^2)

```



```

SSW.complete.c1 = sum (x1.complete)
SSW.complete.c2 = sum (x1.complete,x2.complete)
SSW.complete.c3 = sum (x1.complete,x2.complete,x3.complete)
SSB.complete.c1 = SST - SSW.complete.c1
SSB.complete.c2 = SST - SSW.complete.c2
SSB.complete.c3 = SST - SSW.complete.c3
R.square.complete.c1 = SSB.complete.c1/SST
R.square.complete.c2 = SSB.complete.c2/SST
R.square.complete.c3 = SSB.complete.c3/SST
Hasil.complete = c(SST, SSW.complete.c3,R.square.complete.c3)
# Menghitung sum of square within group (average)
kelompok.average = hasil.kelompok.numerik$average
data.average = data.frame(average, data$X1, data$X2,
data$X3, data$X4, data$X5, data$X6, data$X7, data$X8)
data.average.sort = data.average[order(data.average$average),]
data.c1.average = data.average.sort[1:a,]
data.c2.average = data.average.sort[(a+1):(a+b),]
data.c3.average = data.average.sort[(a+b+1):(a+b+c),]
xlc1bar.average = matrix(c(rep(mean(data.c1.average$data.X1),a),
rep(mean(data.c1.average$data.X2),a),
rep(mean(data.c1.average$data.X3),a),
rep(mean(data.c1.average$data.X4),a),
rep(mean(data.c1.average$data.X5),a),
rep(mean(data.c1.average$data.X6),a),
rep(mean(data.c1.average$data.X7),a),
rep(mean(data.c1.average$data.X8),a)), ncol = 8)
xlc2bar.average = matrix(c(rep(mean(data.c2.average$data.X1),b),
rep(mean(data.c2.average$data.X2),b),
rep(mean(data.c2.average$data.X3),b),
rep(mean(data.c2.average$data.X4),b),
rep(mean(data.c2.average$data.X5),b),
rep(mean(data.c2.average$data.X6),b),
rep(mean(data.c2.average$data.X7),b),
rep(mean(data.c2.average$data.X8),b)), ncol = 8)
xlc3bar.average = matrix(c(rep(mean(data.c3.average$data.X1),c),
rep(mean(data.c3.average$data.X2),c),
rep(mean(data.c3.average$data.X3),c),
rep(mean(data.c3.average$data.X4),c),
rep(mean(data.c3.average$data.X5),c),
rep(mean(data.c3.average$data.X6),c),
rep(mean(data.c3.average$data.X7),c),
rep(mean(data.c3.average$data.X8),c)), ncol = 8)
xil.c1.average =
matrix(c(data.c1.average$data.X1,data.c1.average$data.X2,
data.c1.average$data.X3,data.c1.average$data.X4,
data.c1.average$data.X5,data.c1.average$data.X6,
data.c1.average$data.X7,data.c1.average$data.X8), ncol = 8)
xil.c2.average =
matrix(c(data.c2.average$data.X1,data.c2.average$data.X2,
data.c2.average$data.X3,data.c2.average$data.X4,
data.c2.average$data.X5,data.c2.average$data.X6,
data.c2.average$data.X7,data.c2.average$data.X8), ncol = 8)
xil.c3.average =
matrix(c(data.c3.average$data.X1,data.c3.average$data.X2,
data.c3.average$data.X3,data.c3.average$data.X4,
data.c3.average$data.X5,data.c3.average$data.X6,
data.c3.average$data.X7,data.c3.average$data.X8), ncol = 8)

```

```

x1.average = sum ((xil.c1.average-xlc1bar.average)^2)
x2.average = sum ((xil.c2.average-xlc2bar.average)^2)
x3.average = sum ((xil.c3.average-xlc3bar.average)^2)
SSW.average.c1 = sum (x1.average)
SSW.average.c2 = sum (x1.average,x2.average)
SSW.average.c3 = sum (x1.average,x2.average,x3.average)
SSB.average.c1 = SST - SSW.average.c1
SSB.average.c2 = SST - SSW.average.c2
SSB.average.c3 = SST - SSW.average.c3
R.square.average.c1 = SSB.average.c1/SST
R.square.average.c2 = SSB.average.c2/SST
R.square.average.c3 = SSB.average.c3/SST
Hasil.average = c(SST, SSW.average.c3,R.square.average.c3)
Hasil = rbind(Hasil.single, Hasil.complete, Hasil.average)
print("Hasil Pengelompokan Hirarki")
print(hasil.kelompok.numerik)
print("-----")
print(" Untuk 3 kelompok yang terbentuk ")
print("Metode          SST          SSW          R-square")
print(Hasil)
}

```

c. Jumlah kelompok yang dibentuk sama dengan 4

```

Analisis.kluster.k4 = function(data){
  k = 4
  datanumerik = data.frame(data$X1, data$X2, data$X3, data$X4,
  data$X5, data$X6, data$X7, data$X8)
  d = dist(datanumerik, method = "euclidean")
  # Analisis Kluster Hirarki
  fit.sin = hclust(d, method = "single")
  fit.com = hclust(d, method = "complete")
  fit.ave = hclust(d, method = "average")
  # Memotong Dendrogram untuk k Kluster
  single = cutree(fit.sin, k=k)
  complete = cutree(fit.com, k=k)
  average = cutree(fit.ave, k=k)
  hasil.kelompok.numerik = data.frame>Nama, single, complete,
  average)
  # Menghitung sum of square total  $y = (xil-xlbar)^2$ 
  y = c((data$X1-mean(data$X1))^2, (data$X2-mean(data$X2))^2,
  (data$X3-mean(data$X3))^2, (data$X4-mean(data$X4))^2,
  (data$X5-mean(data$X5))^2, (data$X6-mean(data$X6))^2,
  (data$X7-mean(data$X7))^2, (data$X8-mean(data$X8))^2)
  SST = sum (y)
  # Menghitung sum of square within group (single)
  kelompok.single = hasil.kelompok.numerik$single
  data.single = data.frame(single, data$X1, data$X2,
  data$X3, data$X4, data$X5, data$X6, data$X7, data$X8)
  data.single.sort = data.single[order(data.single$single),]
  data.c1.single = data.single.sort[1:a,]
  data.c2.single = data.single.sort[(a+1):(a+b),]
  data.c3.single = data.single.sort[(a+b+1):(a+b+c),]
  data.c4.single = data.single.sort[(a+b+c+1):(a+b+c+d),]
  xlc1bar.single = matrix(c(rep(mean(data.c1.single$data.X1),a),
  rep(mean(data.c1.single$data.X2),a),
  rep(mean(data.c1.single$data.X3),a),

```

```

rep(mean(data.c1.single$data.X4),a),
rep(mean(data.c1.single$data.X5),a),
rep(mean(data.c1.single$data.X6),a),
rep(mean(data.c1.single$data.X7),a),
rep(mean(data.c1.single$data.X8),a)), ncol = 8)
xlc2bar.single = matrix(c(rep(mean(data.c2.single$data.X1),b),
rep(mean(data.c2.single$data.X2),b),
rep(mean(data.c2.single$data.X3),b),
rep(mean(data.c2.single$data.X4),b),
rep(mean(data.c2.single$data.X5),b),
rep(mean(data.c2.single$data.X6),b),
rep(mean(data.c2.single$data.X7),b),
rep(mean(data.c2.single$data.X8),b)), ncol = 8)
xlc3bar.single = matrix(c(rep(mean(data.c3.single$data.X1),c),
rep(mean(data.c3.single$data.X2),c),
rep(mean(data.c3.single$data.X3),c),
rep(mean(data.c3.single$data.X4),c),
rep(mean(data.c3.single$data.X5),c),
rep(mean(data.c3.single$data.X6),c),
rep(mean(data.c3.single$data.X7),c),
rep(mean(data.c3.single$data.X8),c)), ncol = 8)
xlc4bar.single = matrix(c(rep(mean(data.c4.single$data.X1),d),
rep(mean(data.c4.single$data.X2),d),
rep(mean(data.c4.single$data.X3),d),
rep(mean(data.c4.single$data.X4),d),
rep(mean(data.c4.single$data.X5),d),
rep(mean(data.c4.single$data.X6),d),
rep(mean(data.c4.single$data.X7),d),
rep(mean(data.c4.single$data.X8),d)), ncol = 8)
xil.c1.single =
matrix(c(data.c1.single$data.X1,data.c1.single$data.X2,
data.c1.single$data.X3,data.c1.single$data.X4,
data.c1.single$data.X5,data.c1.single$data.X6,
data.c1.single$data.X7,data.c1.single$data.X8), ncol = 8)
xil.c2.single =
matrix(c(data.c2.single$data.X1,data.c2.single$data.X2,
data.c2.single$data.X3,data.c2.single$data.X4,
data.c2.single$data.X5,data.c2.single$data.X6,
data.c2.single$data.X7,data.c2.single$data.X8), ncol = 8)
xil.c3.single =
matrix(c(data.c3.single$data.X1,data.c3.single$data.X2,
data.c3.single$data.X3,data.c3.single$data.X4,
data.c3.single$data.X5,data.c3.single$data.X6,
data.c3.single$data.X7,data.c3.single$data.X8), ncol = 8)
xil.c4.single =
matrix(c(data.c4.single$data.X1,data.c4.single$data.X2,
data.c4.single$data.X3,data.c4.single$data.X4,
data.c4.single$data.X5,data.c4.single$data.X6,
data.c4.single$data.X7,data.c4.single$data.X8), ncol = 8)
x1.single = sum ((xil.c1.single-xlc1bar.single)^2)
x2.single = sum ((xil.c2.single-xlc2bar.single)^2)
x3.single = sum ((xil.c3.single-xlc3bar.single)^2)
x4.single = sum ((xil.c4.single-xlc4bar.single)^2)
SSW.single.c1 = sum (x1.single)
SSW.single.c2 = sum (x1.single,x2.single)
SSW.single.c3 = sum (x1.single,x2.single,x3.single)
SSW.single.c4 = sum (x1.single,x2.single,x3.single,x4.single)

```

```

SSB.single.c1      = SST - SSW.single.c1
SSB.single.c2      = SST - SSW.single.c2
SSB.single.c3      = SST - SSW.single.c3
SSB.single.c4      = SST - SSW.single.c4
R.square.single.c1 = SSB.single.c1/SST
R.square.single.c2 = SSB.single.c2/SST
R.square.single.c3 = SSB.single.c3/SST
R.square.single.c4 = SSB.single.c4/SST
Hasil.single = c(SST, SSW.single.c4,R.square.single.c4)
# Menghitung sum of square within group (complete)
kelompok.complete      = hasil.kelompok.numerik$complete
data.complete           = data.frame(complete, data$X1, data$X2,
data$X3, data$X4, data$X5, data$X6, data$X7, data$X8)
data.complete.sort      =
data.complete[order(data.complete$complete),]
xlc1bar.complete =
matrix(c(rep(mean(data.c1.complete$data.X1),a),
rep(mean(data.c1.complete$data.X2),a),
rep(mean(data.c1.complete$data.X3),a),
rep(mean(data.c1.complete$data.X4),a),
rep(mean(data.c1.complete$data.X5),a),
rep(mean(data.c1.complete$data.X6),a),
rep(mean(data.c1.complete$data.X7),a),
rep(mean(data.c1.complete$data.X8),a)), ncol = 8)
xlc2bar.complete =
matrix(c(rep(mean(data.c2.complete$data.X1),b),
rep(mean(data.c2.complete$data.X2),b),
rep(mean(data.c2.complete$data.X3),b),
rep(mean(data.c2.complete$data.X4),b),
rep(mean(data.c2.complete$data.X5),b),
rep(mean(data.c2.complete$data.X6),b),
rep(mean(data.c2.complete$data.X7),b),
rep(mean(data.c2.complete$data.X8),b)), ncol = 8)
xlc3bar.complete =
matrix(c(rep(mean(data.c3.complete$data.X1),c),
rep(mean(data.c3.complete$data.X2),c),
rep(mean(data.c3.complete$data.X3),c),
rep(mean(data.c3.complete$data.X4),c),
rep(mean(data.c3.complete$data.X5),c),
rep(mean(data.c3.complete$data.X6),c),
rep(mean(data.c3.complete$data.X7),c),
rep(mean(data.c3.complete$data.X8),c)), ncol = 8)
xlc4bar.complete =
matrix(c(rep(mean(data.c4.complete$data.X1),d),
rep(mean(data.c4.complete$data.X2),d),
rep(mean(data.c4.complete$data.X3),d),
rep(mean(data.c4.complete$data.X4),d),
rep(mean(data.c4.complete$data.X5),d),
rep(mean(data.c4.complete$data.X6),d),
rep(mean(data.c4.complete$data.X7),d),
rep(mean(data.c4.complete$data.X8),d)), ncol = 8)
xil.c1.complete =
matrix(c(data.c1.complete$data.X1,data.c1.complete$data.X2,
data.c1.complete$data.X3,data.c1.complete$data.X4,
data.c1.complete$data.X5,data.c1.complete$data.X6,
data.c1.complete$data.X7,data.c1.complete$data.X8), ncol = 8)

```

```

xil.c2.complete =
matrix(c(data.c2.complete$data.X1,data.c2.complete$data.X2,
data.c2.complete$data.X3,data.c2.complete$data.X4,
data.c2.complete$data.X5,data.c2.complete$data.X6,
data.c2.complete$data.X7,data.c2.complete$data.X8), ncol = 8)
xil.c3.complete =
matrix(c(data.c3.complete$data.X1,data.c3.complete$data.X2,
data.c3.complete$data.X3,data.c3.complete$data.X4,
data.c3.complete$data.X5,data.c3.complete$data.X6,
data.c3.complete$data.X7,data.c3.complete$data.X8), ncol = 8)
xil.c4.complete =
matrix(c(data.c4.complete$data.X1,data.c4.complete$data.X2,
data.c4.complete$data.X3,data.c4.complete$data.X4,
data.c4.complete$data.X5,data.c4.complete$data.X6,
data.c4.complete$data.X7,data.c4.complete$data.X8), ncol = 8)
x1.complete = sum ((xil.c1.complete-xlc1bar.complete)^2)
x2.complete = sum ((xil.c2.complete-xlc2bar.complete)^2)
x3.complete = sum ((xil.c3.complete-xlc3bar.complete)^2)
x4.complete = sum ((xil.c4.complete-xlc4bar.complete)^2)
SSW.complete.c1 = sum (x1.complete)
SSW.complete.c2 = sum (x1.complete,x2.complete)
SSW.complete.c3 = sum (x1.complete,x2.complete,x3.complete)
SSW.complete.c4 = sum
(x1.complete,x2.complete,x3.complete,x4.complete)
SSB.complete.c1 = SST - SSW.complete.c1
SSB.complete.c2 = SST - SSW.complete.c2
SSB.complete.c3 = SST - SSW.complete.c3
SSB.complete.c4 = SST - SSW.complete.c4
R.square.complete.c1 = SSB.complete.c1/SST
R.square.complete.c2 = SSB.complete.c2/SST
R.square.complete.c3 = SSB.complete.c3/SST
R.square.complete.c4 = SSB.complete.c4/SST
Hasil.complete = c(SST, SSW.complete.c4,R.square.complete.c4)
# Menghitung sum of square within group (average)
kelompok.average = hasil.kelompok.numerik$average
data.average = data.frame(average, data$X1, data$X2,
data$X3, data$X4, data$X5, data$X6, data$X7, data$X8)
data.average.sort = data.average[order(data.average$average),]
xlc1bar.average = matrix(c(rep(mean(data.c1.average$data.X1),a),
rep(mean(data.c1.average$data.X2),a),
rep(mean(data.c1.average$data.X3),a),
rep(mean(data.c1.average$data.X4),a),
rep(mean(data.c1.average$data.X5),a),
rep(mean(data.c1.average$data.X6),a),
rep(mean(data.c1.average$data.X7),a),
rep(mean(data.c1.average$data.X8),a)), ncol = 8)
xlc2bar.average = matrix(c(rep(mean(data.c2.average$data.X1),b),
rep(mean(data.c2.average$data.X2),b),
rep(mean(data.c2.average$data.X3),b),
rep(mean(data.c2.average$data.X4),b),
rep(mean(data.c2.average$data.X5),b),
rep(mean(data.c2.average$data.X6),b),
rep(mean(data.c2.average$data.X7),b),
rep(mean(data.c2.average$data.X8),b)), ncol = 8)
xlc3bar.average = matrix(c(rep(mean(data.c3.average$data.X1),c),
rep(mean(data.c3.average$data.X2),c),
rep(mean(data.c3.average$data.X3),c),

```

```

rep(mean(data.c3.average$data.X4),c),
rep(mean(data.c3.average$data.X5),c),
rep(mean(data.c3.average$data.X6),c),
rep(mean(data.c3.average$data.X7),c),
rep(mean(data.c3.average$data.X8),c)), ncol = 8)
xlc4bar.average = matrix(c(rep(mean(data.c4.average$data.X1),d),
rep(mean(data.c4.average$data.X2),d),
rep(mean(data.c4.average$data.X3),d),
rep(mean(data.c4.average$data.X4),d),
rep(mean(data.c4.average$data.X5),d),
rep(mean(data.c4.average$data.X6),d),
rep(mean(data.c4.average$data.X7),d),
rep(mean(data.c4.average$data.X8),d)), ncol = 8)
xil.c1.average =
matrix(c(data.c1.average$data.X1,data.c1.average$data.X2,
data.c1.average$data.X3,data.c1.average$data.X4,
data.c1.average$data.X5,data.c1.average$data.X6,
data.c1.average$data.X7,data.c1.average$data.X8), ncol = 8)
xil.c2.average =
matrix(c(data.c2.average$data.X1,data.c2.average$data.X2,
data.c2.average$data.X3,data.c2.average$data.X4,
data.c2.average$data.X5,data.c2.average$data.X6,
data.c2.average$data.X7,data.c2.average$data.X8), ncol = 8)
xil.c3.average =
matrix(c(data.c3.average$data.X1,data.c3.average$data.X2,
data.c3.average$data.X3,data.c3.average$data.X4,
data.c3.average$data.X5,data.c3.average$data.X6,
data.c3.average$data.X7,data.c3.average$data.X8), ncol = 8)
xil.c4.average =
matrix(c(data.c4.average$data.X1,data.c4.average$data.X2,
data.c4.average$data.X3,data.c4.average$data.X4,
data.c4.average$data.X5,data.c4.average$data.X6,
data.c4.average$data.X7,data.c4.average$data.X8), ncol = 8)
x1.average = sum ((xil.c1.average-xlc1bar.average)^2)
x2.average = sum ((xil.c2.average-xlc2bar.average)^2)
x3.average = sum ((xil.c3.average-xlc3bar.average)^2)
x4.average = sum ((xil.c4.average-xlc4bar.average)^2)
SSW.average.c1 = sum (x1.average)
SSW.average.c2 = sum (x1.average,x2.average)
SSW.average.c3 = sum (x1.average,x2.average,x3.average)
SSW.average.c4 = sum
(x1.average,x2.average,x3.average,x4.average)
SSB.average.c1 = SST - SSW.average.c1
SSB.average.c2 = SST - SSW.average.c2
SSB.average.c3 = SST - SSW.average.c3
SSB.average.c4 = SST - SSW.average.c4
R.square.average.c1 = SSB.average.c1/SST
R.square.average.c2 = SSB.average.c2/SST
R.square.average.c3 = SSB.average.c3/SST
R.square.average.c4 = SSB.average.c4/SST
Hasil.average = c(SST, SSW.average.c4,R.square.average.c4)
Hasil = rbind(Hasil.single, Hasil.complete, Hasil.average)
print("Hasil Pengelompokan Hirarki")
print(hasil.kelompok.numerik)
print("-----")
print(" Untuk 4 kelompok yang terbentuk ")
print("Metode SST SSW R-square")

```

```
print(Hasil)
}
```

d. Jumlah kelompok yang dibentuk sama dengan 5

```
Analisis.kluster.k5 = function(data){
  k
    = 5
  datanumerik = data.frame(data$X1, data$X2, data$X3, data$X4,
    data$X5, data$X6, data$X7, data$X8)
  d
    = dist(datanumerik, method = "euclidean")
  # Analisis Kluster Hirarki
  fit.sin
    = hclust(d, method = "single")
  fit.com
    = hclust(d, method = "complete")
  fit.ave
    = hclust(d, method = "average")
  # Memotong Dendrogram untuk k Kluster
  single
    = cutree(fit.sin, k=k)
  complete
    = cutree(fit.com, k=k)
  average
    = cutree(fit.ave, k=k)
  hasil.kelompok.numerik = data.frame>Nama, single, complete,
    average)
  # Menghitung sum of square total  $y = (x_{il} - \bar{x}_l)^2$ 
  y = c((data$X1-mean(data$X1))^2, (data$X2-mean(data$X2))^2,
    (data$X3-mean(data$X3))^2, (data$X4-mean(data$X4))^2,
    (data$X5-mean(data$X5))^2, (data$X6-mean(data$X6))^2,
    (data$X7-mean(data$X7))^2, (data$X8-mean(data$X8))^2)
  SST
    = sum (y)
  # Menghitung sum of square within group (single)
  kelompok.single
    = hasil.kelompok.numerik$single
  data.single
    = data.frame(single, data$X1, data$X2,
    data$X3, data$X4, data$X5, data$X6, data$X7, data$X8)
  data.single.sort
    = data.single[order(data.single$single),]
  xlc1bar.single = matrix(c(rep(mean(data.c1.single$data.X1),a),
    rep(mean(data.c1.single$data.X2),a),
    rep(mean(data.c1.single$data.X3),a),
    rep(mean(data.c1.single$data.X4),a),
    rep(mean(data.c1.single$data.X5),a),
    rep(mean(data.c1.single$data.X6),a),
    rep(mean(data.c1.single$data.X7),a),
    rep(mean(data.c1.single$data.X8),a)), ncol = 8)
  xlc2bar.single = matrix(c(rep(mean(data.c2.single$data.X1),b),
    rep(mean(data.c2.single$data.X2),b),
    rep(mean(data.c2.single$data.X3),b),
    rep(mean(data.c2.single$data.X4),b),
    rep(mean(data.c2.single$data.X5),b),
    rep(mean(data.c2.single$data.X6),b),
    rep(mean(data.c2.single$data.X7),b),
    rep(mean(data.c2.single$data.X8),b)), ncol = 8)
  xlc3bar.single = matrix(c(rep(mean(data.c3.single$data.X1),c),
    rep(mean(data.c3.single$data.X2),c),
    rep(mean(data.c3.single$data.X3),c),
    rep(mean(data.c3.single$data.X4),c),
    rep(mean(data.c3.single$data.X5),c),
    rep(mean(data.c3.single$data.X6),c),
    rep(mean(data.c3.single$data.X7),c),
    rep(mean(data.c3.single$data.X8),c)), ncol = 8)
  xlc4bar.single = matrix(c(rep(mean(data.c4.single$data.X1),d),
```

```

rep(mean(data.c4.single$data.X2),d),
rep(mean(data.c4.single$data.X3),d),
rep(mean(data.c4.single$data.X4),d),
rep(mean(data.c4.single$data.X5),d),
rep(mean(data.c4.single$data.X6),d),
rep(mean(data.c4.single$data.X7),d),
rep(mean(data.c4.single$data.X8),d)), ncol = 8)
xlc5bar.single = matrix(c(rep(mean(data.c5.single$data.X1),e),
rep(mean(data.c5.single$data.X2),e),
rep(mean(data.c5.single$data.X3),e),
rep(mean(data.c5.single$data.X4),e),
rep(mean(data.c5.single$data.X5),e),
rep(mean(data.c5.single$data.X6),e),
rep(mean(data.c5.single$data.X7),e),
rep(mean(data.c5.single$data.X8),e))), ncol = 8)
xil.c1.single =
matrix(c(data.c1.single$data.X1,data.c1.single$data.X2,
data.c1.single$data.X3,data.c1.single$data.X4,
data.c1.single$data.X5,data.c1.single$data.X6,
data.c1.single$data.X7,data.c1.single$data.X8), ncol = 8)
xil.c2.single =
matrix(c(data.c2.single$data.X1,data.c2.single$data.X2,
data.c2.single$data.X3,data.c2.single$data.X4,
data.c2.single$data.X5,data.c2.single$data.X6,
data.c2.single$data.X7,data.c2.single$data.X8), ncol = 8)
xil.c3.single =
matrix(c(data.c3.single$data.X1,data.c3.single$data.X2,
data.c3.single$data.X3,data.c3.single$data.X4,
data.c3.single$data.X5,data.c3.single$data.X6,
data.c3.single$data.X7,data.c3.single$data.X8), ncol = 8)
xil.c4.single =
matrix(c(data.c4.single$data.X1,data.c4.single$data.X2,
data.c4.single$data.X3,data.c4.single$data.X4,
data.c4.single$data.X5,data.c4.single$data.X6,
data.c4.single$data.X7,data.c4.single$data.X8), ncol = 8)
xil.c5.single =
matrix(c(data.c5.single$data.X1,data.c5.single$data.X2,
data.c5.single$data.X3,data.c5.single$data.X4,
data.c5.single$data.X5,data.c5.single$data.X6,
data.c5.single$data.X7,data.c5.single$data.X8), ncol = 8)
x1.single = sum ((xil.c1.single-xlc1bar.single)^2)
x2.single = sum ((xil.c2.single-xlc2bar.single)^2)
x3.single = sum ((xil.c3.single-xlc3bar.single)^2)
x4.single = sum ((xil.c4.single-xlc4bar.single)^2)
x5.single = sum ((xil.c5.single-xlc5bar.single)^2)
SSW.single.c1 = sum (x1.single)
SSW.single.c2 = sum (x1.single,x2.single)
SSW.single.c3 = sum (x1.single,x2.single,x3.single)
SSW.single.c4 = sum (x1.single,x2.single,x3.single,x4.single)
SSW.single.c5 = sum
(x1.single,x2.single,x3.single,x4.single,x5.single)
SSB.single.c1 = SST - SSW.single.c1
SSB.single.c2 = SST - SSW.single.c2
SSB.single.c3 = SST - SSW.single.c3
SSB.single.c4 = SST - SSW.single.c4
SSB.single.c5 = SST - SSW.single.c5
R.square.single.c1 = SSB.single.c1/SST

```



```

R.square.single.c2 = SSB.single.c2/SST
R.square.single.c3 = SSB.single.c3/SST
R.square.single.c4 = SSB.single.c4/SST
R.square.single.c5 = SSB.single.c5/SST
Hasil.single = c(SST, SSW.single.c5,R.square.single.c5)
# Menghitung sum of square within group (complete)
kelompok.complete = hasil.kelompok.numerik$complete
data.complete = data.frame(complete, data$X1, data$X2,
data$X3, data$X4, data$X5, data$X6, data$X7, data$X8)
data.complete.sort=
data.complete[order(data.complete$complete),]
xlc1bar.complete =
matrix(c(rep(mean(data.c1.complete$data.X1),a),
rep(mean(data.c1.complete$data.X2),a),
rep(mean(data.c1.complete$data.X3),a),
rep(mean(data.c1.complete$data.X4),a),
rep(mean(data.c1.complete$data.X5),a),
rep(mean(data.c1.complete$data.X6),a),
rep(mean(data.c1.complete$data.X7),a),
rep(mean(data.c1.complete$data.X8),a)), ncol = 8)
xlc2bar.complete =
matrix(c(rep(mean(data.c2.complete$data.X1),b),
rep(mean(data.c2.complete$data.X2),b),
rep(mean(data.c2.complete$data.X3),b),
rep(mean(data.c2.complete$data.X4),b),
rep(mean(data.c2.complete$data.X5),b),
rep(mean(data.c2.complete$data.X6),b),
rep(mean(data.c2.complete$data.X7),b),
rep(mean(data.c2.complete$data.X8),b)), ncol = 8)
xlc3bar.complete =
matrix(c(rep(mean(data.c3.complete$data.X1),c),
rep(mean(data.c3.complete$data.X2),c),
rep(mean(data.c3.complete$data.X3),c),
rep(mean(data.c3.complete$data.X4),c),
rep(mean(data.c3.complete$data.X5),c),
rep(mean(data.c3.complete$data.X6),c),
rep(mean(data.c3.complete$data.X7),c),
rep(mean(data.c3.complete$data.X8),c)), ncol = 8)
xlc4bar.complete =
matrix(c(rep(mean(data.c4.complete$data.X1),d),
rep(mean(data.c4.complete$data.X2),d),
rep(mean(data.c4.complete$data.X3),d),
rep(mean(data.c4.complete$data.X4),d),
rep(mean(data.c4.complete$data.X5),d),
rep(mean(data.c4.complete$data.X6),d),
rep(mean(data.c4.complete$data.X7),d),
rep(mean(data.c4.complete$data.X8),d)), ncol = 8)
xlc5bar.complete =
matrix(c(rep(mean(data.c5.complete$data.X1),e),
rep(mean(data.c5.complete$data.X2),e),
rep(mean(data.c5.complete$data.X3),e),
rep(mean(data.c5.complete$data.X4),e),
rep(mean(data.c5.complete$data.X5),e),
rep(mean(data.c5.complete$data.X6),e),
rep(mean(data.c5.complete$data.X7),e),
rep(mean(data.c5.complete$data.X8),e)), ncol = 8)

```

```

xil.c1.complete =
matrix(c(data.c1.complete$data.X1,data.c1.complete$data.X2,
data.c1.complete$data.X3,data.c1.complete$data.X4,
data.c1.complete$data.X5,data.c1.complete$data.X6,
data.c1.complete$data.X7,data.c1.complete$data.X8), ncol = 8)
xil.c2.complete =
matrix(c(data.c2.complete$data.X1,data.c2.complete$data.X2,
data.c2.complete$data.X3,data.c2.complete$data.X4,
data.c2.complete$data.X5,data.c2.complete$data.X6,
data.c2.complete$data.X7,data.c2.complete$data.X8), ncol = 8)
xil.c3.complete =
matrix(c(data.c3.complete$data.X1,data.c3.complete$data.X2,
data.c3.complete$data.X3,data.c3.complete$data.X4,
data.c3.complete$data.X5,data.c3.complete$data.X6,
data.c3.complete$data.X7,data.c3.complete$data.X8), ncol = 8)
xil.c4.complete =
matrix(c(data.c4.complete$data.X1,data.c4.complete$data.X2,
data.c4.complete$data.X3,data.c4.complete$data.X4,
data.c4.complete$data.X5,data.c4.complete$data.X6,
data.c4.complete$data.X7,data.c4.complete$data.X8), ncol = 8)
xil.c5.complete =
matrix(c(data.c5.complete$data.X1,data.c5.complete$data.X2,
data.c5.complete$data.X3,data.c5.complete$data.X4,
data.c5.complete$data.X5,data.c5.complete$data.X6,
data.c5.complete$data.X7,data.c5.complete$data.X8), ncol = 8)
x1.complete = sum ((xil.c1.complete-xlc1bar.complete)^2)
x2.complete = sum ((xil.c2.complete-xlc2bar.complete)^2)
x3.complete = sum ((xil.c3.complete-xlc3bar.complete)^2)
x4.complete = sum ((xil.c4.complete-xlc4bar.complete)^2)
x5.complete = sum ((xil.c5.complete-xlc5bar.complete)^2)
SSW.complete.c1 = sum (x1.complete)
SSW.complete.c2 = sum (x1.complete,x2.complete)
SSW.complete.c3 = sum (x1.complete,x2.complete,x3.complete)
SSW.complete.c4 = sum
(x1.complete,x2.complete,x3.complete,x4.complete)
SSW.complete.c5 = sum
(x1.complete,x2.complete,x3.complete,x4.complete,x5.complete)
SSB.complete.c1 = SST - SSW.complete.c1
SSB.complete.c2 = SST - SSW.complete.c2
SSB.complete.c3 = SST - SSW.complete.c3
SSB.complete.c4 = SST - SSW.complete.c4
SSB.complete.c5 = SST - SSW.complete.c5
R.square.complete.c1 = SSB.complete.c1/SST
R.square.complete.c2 = SSB.complete.c2/SST
R.square.complete.c3 = SSB.complete.c3/SST
R.square.complete.c4 = SSB.complete.c4/SST
R.square.complete.c5 = SSB.complete.c5/SST
Hasil.complete = c(SST, SSW.complete.c5,R.square.complete.c5)
# Menghitung sum of square within group (average)
kelompok.average = hasil.kelompok.numerik$average
data.average = data.frame(average, data$X1, data$X2,
data$X3, data$X4, data$X5, data$X6, data$X7, data$X8)
data.average.sort = data.average[order(data.average$average),]
xlc1bar.average = matrix(c(rep(mean(data.c1.average$data.X1),a),
rep(mean(data.c1.average$data.X2),a),
rep(mean(data.c1.average$data.X3),a),
rep(mean(data.c1.average$data.X4),a),

```

```

rep(mean(data.c1.average$data.X5),a),
rep(mean(data.c1.average$data.X6),a),
rep(mean(data.c1.average$data.X7),a),
rep(mean(data.c1.average$data.X8),a)), ncol = 8)
xlc2bar.average = matrix(c(rep(mean(data.c2.average$data.X1),b),
rep(mean(data.c2.average$data.X2),b),
rep(mean(data.c2.average$data.X3),b),
rep(mean(data.c2.average$data.X4),b),
rep(mean(data.c2.average$data.X5),b),
rep(mean(data.c2.average$data.X6),b),
rep(mean(data.c2.average$data.X7),b),
rep(mean(data.c2.average$data.X8),b))), ncol = 8)
xlc3bar.average = matrix(c(rep(mean(data.c3.average$data.X1),c),
rep(mean(data.c3.average$data.X2),c),
rep(mean(data.c3.average$data.X3),c),
rep(mean(data.c3.average$data.X4),c),
rep(mean(data.c3.average$data.X5),c),
rep(mean(data.c3.average$data.X6),c),
rep(mean(data.c3.average$data.X7),c),
rep(mean(data.c3.average$data.X8),c))), ncol = 8)
xlc4bar.average = matrix(c(rep(mean(data.c4.average$data.X1),d),
rep(mean(data.c4.average$data.X2),d),
rep(mean(data.c4.average$data.X3),d),
rep(mean(data.c4.average$data.X4),d),
rep(mean(data.c4.average$data.X5),d),
rep(mean(data.c4.average$data.X6),d),
rep(mean(data.c4.average$data.X7),d),
rep(mean(data.c4.average$data.X8),d))), ncol = 8)
xlc5bar.average = matrix(c(rep(mean(data.c5.average$data.X1),e),
rep(mean(data.c5.average$data.X2),e),
rep(mean(data.c5.average$data.X3),e),
rep(mean(data.c5.average$data.X4),e),
rep(mean(data.c5.average$data.X5),e),
rep(mean(data.c5.average$data.X6),e),
rep(mean(data.c5.average$data.X7),e),
rep(mean(data.c5.average$data.X8),e))), ncol = 8)
xil.c1.average =
matrix(c(data.c1.average$data.X1,data.c1.average$data.X2,
data.c1.average$data.X3,data.c1.average$data.X4,
data.c1.average$data.X5,data.c1.average$data.X6,
data.c1.average$data.X7,data.c1.average$data.X8), ncol = 8)
xil.c2.average =
matrix(c(data.c2.average$data.X1,data.c2.average$data.X2,
data.c2.average$data.X3,data.c2.average$data.X4,
data.c2.average$data.X5,data.c2.average$data.X6,
data.c2.average$data.X7,data.c2.average$data.X8), ncol = 8)
xil.c3.average =
matrix(c(data.c3.average$data.X1,data.c3.average$data.X2,
data.c3.average$data.X3,data.c3.average$data.X4,
data.c3.average$data.X5,data.c3.average$data.X6,
data.c3.average$data.X7,data.c3.average$data.X8), ncol = 8)
xil.c4.average =
matrix(c(data.c4.average$data.X1,data.c4.average$data.X2,
data.c4.average$data.X3,data.c4.average$data.X4,
data.c4.average$data.X5,data.c4.average$data.X6,
data.c4.average$data.X7,data.c4.average$data.X8), ncol = 8)

```

```

xil.c5.average =
matrix(c(data.c5.average$data.X1,data.c5.average$data.X2,
data.c5.average$data.X3,data.c5.average$data.X4,
data.c5.average$data.X5,data.c5.average$data.X6,
data.c5.average$data.X7,data.c5.average$data.X8), ncol = 8)
x1.average = sum ((xil.c1.average-xlc1bar.average)^2)
x2.average = sum ((xil.c2.average-xlc2bar.average)^2)
x3.average = sum ((xil.c3.average-xlc3bar.average)^2)
x4.average = sum ((xil.c4.average-xlc4bar.average)^2)
x5.average = sum ((xil.c5.average-xlc5bar.average)^2)
SSW.average.c1 = sum (x1.average)
SSW.average.c2 = sum (x1.average,x2.average)
SSW.average.c3 = sum (x1.average,x2.average,x3.average)
SSW.average.c4 = sum
(x1.average,x2.average,x3.average,x4.average)
SSW.average.c5 = sum
(x1.average,x2.average,x3.average,x4.average,x5.average)
SSB.average.c1 = SST - SSW.average.c1
SSB.average.c2 = SST - SSW.average.c2
SSB.average.c3 = SST - SSW.average.c3
SSB.average.c4 = SST - SSW.average.c4
SSB.average.c5 = SST - SSW.average.c5
R.square.average.c1 = SSB.average.c1/SST
R.square.average.c2 = SSB.average.c2/SST
R.square.average.c3 = SSB.average.c3/SST
R.square.average.c4 = SSB.average.c4/SST
R.square.average.c5 = SSB.average.c5/SST
Hasil.average = c(SST, SSW.average.c5,R.square.average.c5)
Hasil = rbind(Hasil.single, Hasil.complete, Hasil.average)
print("Hasil Pengelompokan Hirarki")
print(hasil.kelompok.numerik)
print("-----")
print(" Untuk 5 kelompok yang terbentuk ")
print("Metode          SST          SSW          R-square")
print(Hasil)
}

```

e. Jumlah kelompok yang dibentuk sama dengan 6

```

Analisis.kluster.k6 = function(data){
k
= 6
datanumerik = data.frame(data$X1, data$X2, data$X3, data$X4,
data$X5, data$X6, data$X7, data$X8)
d
= dist(datanumerik, method = "euclidean")
# Analisis Kluster Hirarki
fit.sin
= hclust(d, method = "single")
fit.com
= hclust(d, method = "complete")
fit.ave
= hclust(d, method = "average")
# Memotong Dendrogram untuk k Kluster
single
= cutree(fit.sin, k=k)
complete
= cutree(fit.com, k=k)
average
= cutree(fit.ave, k=k)
hasil.kelompok.numerik = data.frame>Nama, single, complete,
average)
# Menghitung sum of square total  $y = (xil - \bar{x})^2$ 
y = c((data$X1-mean(data$X1))^2, (data$X2-mean(data$X2))^2,
(data$X3-mean(data$X3))^2, (data$X4-mean(data$X4))^2,

```

```

(data$X5-mean(data$X5))^2, (data$X6-mean(data$X6))^2,
(data$X7-mean(data$X7))^2, (data$X8-mean(data$X8))^2)
SST = sum (y)
# Menghitung sum of square within group (single)
kelompok.single = hasil.kelompok.numerik$single
data.single = data.frame(single, data$X1, data$X2,
data$X3, data$X4, data$X5, data$X6, data$X7, data$X8)
data.single.sort = data.single[order(data.single$single),]
xlc1bar.single = matrix(c(rep(mean(data.c1.single$data.X1),a),
rep(mean(data.c1.single$data.X2),a),
rep(mean(data.c1.single$data.X3),a),
rep(mean(data.c1.single$data.X4),a),
rep(mean(data.c1.single$data.X5),a),
rep(mean(data.c1.single$data.X6),a),
rep(mean(data.c1.single$data.X7),a),
rep(mean(data.c1.single$data.X8),a)), ncol = 8)
xlc2bar.single = matrix(c(rep(mean(data.c2.single$data.X1),b),
rep(mean(data.c2.single$data.X2),b),
rep(mean(data.c2.single$data.X3),b),
rep(mean(data.c2.single$data.X4),b),
rep(mean(data.c2.single$data.X5),b),
rep(mean(data.c2.single$data.X6),b),
rep(mean(data.c2.single$data.X7),b),
rep(mean(data.c2.single$data.X8),b)), ncol = 8)
xlc3bar.single = matrix(c(rep(mean(data.c3.single$data.X1),c),
rep(mean(data.c3.single$data.X2),c),
rep(mean(data.c3.single$data.X3),c),
rep(mean(data.c3.single$data.X4),c),
rep(mean(data.c3.single$data.X5),c),
rep(mean(data.c3.single$data.X6),c),
rep(mean(data.c3.single$data.X7),c),
rep(mean(data.c3.single$data.X8),c)), ncol = 8)
xlc4bar.single = matrix(c(rep(mean(data.c4.single$data.X1),d),
rep(mean(data.c4.single$data.X2),d),
rep(mean(data.c4.single$data.X3),d),
rep(mean(data.c4.single$data.X4),d),
rep(mean(data.c4.single$data.X5),d),
rep(mean(data.c4.single$data.X6),d),
rep(mean(data.c4.single$data.X7),d),
rep(mean(data.c4.single$data.X8),d)), ncol = 8)
xlc5bar.single = matrix(c(rep(mean(data.c5.single$data.X1),e),
rep(mean(data.c5.single$data.X2),e),
rep(mean(data.c5.single$data.X3),e),
rep(mean(data.c5.single$data.X4),e),
rep(mean(data.c5.single$data.X5),e),
rep(mean(data.c5.single$data.X6),e),
rep(mean(data.c5.single$data.X7),e),
rep(mean(data.c5.single$data.X8),e)), ncol = 8)
xlc6bar.single = matrix(c(rep(mean(data.c6.single$data.X1),f),
rep(mean(data.c6.single$data.X2),f),
rep(mean(data.c6.single$data.X3),f),
rep(mean(data.c6.single$data.X4),f),
rep(mean(data.c6.single$data.X5),f),
rep(mean(data.c6.single$data.X6),f),
rep(mean(data.c6.single$data.X7),f),
rep(mean(data.c6.single$data.X8),f)), ncol = 8)

```

```

xil.c1.single =
matrix(c(data.c1.single$data.X1,data.c1.single$data.X2,
data.c1.single$data.X3,data.c1.single$data.X4,
data.c1.single$data.X5,data.c1.single$data.X6,
data.c1.single$data.X7,data.c1.single$data.X8), ncol = 8)
xil.c2.single =
matrix(c(data.c2.single$data.X1,data.c2.single$data.X2,
data.c2.single$data.X3,data.c2.single$data.X4,
data.c2.single$data.X5,data.c2.single$data.X6,
data.c2.single$data.X7,data.c2.single$data.X8), ncol = 8)
xil.c3.single =
matrix(c(data.c3.single$data.X1,data.c3.single$data.X2,
data.c3.single$data.X3,data.c3.single$data.X4,
data.c3.single$data.X5,data.c3.single$data.X6,
data.c3.single$data.X7,data.c3.single$data.X8), ncol = 8)
xil.c4.single =
matrix(c(data.c4.single$data.X1,data.c4.single$data.X2,
data.c4.single$data.X3,data.c4.single$data.X4,
data.c4.single$data.X5,data.c4.single$data.X6,
data.c4.single$data.X7,data.c4.single$data.X8), ncol = 8)
xil.c5.single =
matrix(c(data.c5.single$data.X1,data.c5.single$data.X2,
data.c5.single$data.X3,data.c5.single$data.X4,
data.c5.single$data.X5,data.c5.single$data.X6,
data.c5.single$data.X7,data.c5.single$data.X8), ncol = 8)
xil.c6.single =
matrix(c(data.c6.single$data.X1,data.c6.single$data.X2,
data.c6.single$data.X3,data.c6.single$data.X4,
data.c6.single$data.X5,data.c6.single$data.X6,
data.c6.single$data.X7,data.c6.single$data.X8), ncol = 8)
x1.single = sum ((xil.c1.single-xlc1bar.single)^2)
x2.single = sum ((xil.c2.single-xlc2bar.single)^2)
x3.single = sum ((xil.c3.single-xlc3bar.single)^2)
x4.single = sum ((xil.c4.single-xlc4bar.single)^2)
x5.single = sum ((xil.c5.single-xlc5bar.single)^2)
x6.single = sum ((xil.c6.single-xlc6bar.single)^2)
SSW.single.c1 = sum (x1.single)
SSW.single.c2 = sum (x1.single,x2.single)
SSW.single.c3 = sum (x1.single,x2.single,x3.single)
SSW.single.c4 = sum (x1.single,x2.single,x3.single,x4.single)
SSW.single.c5 = sum
(x1.single,x2.single,x3.single,x4.single,x5.single)
SSW.single.c6 = sum
(x1.single,x2.single,x3.single,x4.single,x5.single,x6.single)
SSB.single.c1 = SST - SSW.single.c1
SSB.single.c2 = SST - SSW.single.c2
SSB.single.c3 = SST - SSW.single.c3
SSB.single.c4 = SST - SSW.single.c4
SSB.single.c5 = SST - SSW.single.c5
SSB.single.c6 = SST - SSW.single.c6
R.square.single.c1 = SSB.single.c1/SST
R.square.single.c2 = SSB.single.c2/SST
R.square.single.c3 = SSB.single.c3/SST
R.square.single.c4 = SSB.single.c4/SST
R.square.single.c5 = SSB.single.c5/SST
R.square.single.c6 = SSB.single.c6/SST
Hasil.single = c(SST, SSW.single.c6,R.square.single.c6)

```

```

# Menghitung sum of square within group (complete)
kelompok.complete = hasil.kelompok.numerik$complete
data.complete = data.frame(complete, data$X1, data$X2,
data$X3, data$X4, data$X5, data$X6, data$X7, data$X8)
data.complete.sort =
data.complete[order(data.complete$complete),]
xlc1bar.complete =
matrix(c(rep(mean(data.c1.complete$data.X1),a),
rep(mean(data.c1.complete$data.X2),a),
rep(mean(data.c1.complete$data.X3),a),
rep(mean(data.c1.complete$data.X4),a),
rep(mean(data.c1.complete$data.X5),a),
rep(mean(data.c1.complete$data.X6),a),
rep(mean(data.c1.complete$data.X7),a),
rep(mean(data.c1.complete$data.X8),a)), ncol = 8)
xlc2bar.complete =
matrix(c(rep(mean(data.c2.complete$data.X1),b),
rep(mean(data.c2.complete$data.X2),b),
rep(mean(data.c2.complete$data.X3),b),
rep(mean(data.c2.complete$data.X4),b),
rep(mean(data.c2.complete$data.X5),b),
rep(mean(data.c2.complete$data.X6),b),
rep(mean(data.c2.complete$data.X7),b),
rep(mean(data.c2.complete$data.X8),b)), ncol = 8)
xlc3bar.complete =
matrix(c(rep(mean(data.c3.complete$data.X1),c),
rep(mean(data.c3.complete$data.X2),c),
rep(mean(data.c3.complete$data.X3),c),
rep(mean(data.c3.complete$data.X4),c),
rep(mean(data.c3.complete$data.X5),c),
rep(mean(data.c3.complete$data.X6),c),
rep(mean(data.c3.complete$data.X7),c),
rep(mean(data.c3.complete$data.X8),c)), ncol = 8)
xlc4bar.complete =
matrix(c(rep(mean(data.c4.complete$data.X1),d),
rep(mean(data.c4.complete$data.X2),d),
rep(mean(data.c4.complete$data.X3),d),
rep(mean(data.c4.complete$data.X4),d),
rep(mean(data.c4.complete$data.X5),d),
rep(mean(data.c4.complete$data.X6),d),
rep(mean(data.c4.complete$data.X7),d),
rep(mean(data.c4.complete$data.X8),d)), ncol = 8)
xlc5bar.complete =
matrix(c(rep(mean(data.c5.complete$data.X1),e),
rep(mean(data.c5.complete$data.X2),e),
rep(mean(data.c5.complete$data.X3),e),
rep(mean(data.c5.complete$data.X4),e),
rep(mean(data.c5.complete$data.X5),e),
rep(mean(data.c5.complete$data.X6),e),
rep(mean(data.c5.complete$data.X7),e),
rep(mean(data.c5.complete$data.X8),e)), ncol = 8)
xlc6bar.complete =
matrix(c(rep(mean(data.c6.complete$data.X1),f),
rep(mean(data.c6.complete$data.X2),f),
rep(mean(data.c6.complete$data.X3),f),
rep(mean(data.c6.complete$data.X4),f),
rep(mean(data.c6.complete$data.X5),f),

```

```

rep(mean(data.c6.complete$data.X6),f),
rep(mean(data.c6.complete$data.X7),f),
rep(mean(data.c6.complete$data.X8),f)), ncol = 8)
xil.c1.complete =
matrix(c(data.c1.complete$data.X1,data.c1.complete$data.X2,
data.c1.complete$data.X3,data.c1.complete$data.X4,
data.c1.complete$data.X5,data.c1.complete$data.X6,
data.c1.complete$data.X7,data.c1.complete$data.X8), ncol = 8)
xil.c2.complete =
matrix(c(data.c2.complete$data.X1,data.c2.complete$data.X2,
data.c2.complete$data.X3,data.c2.complete$data.X4,
data.c2.complete$data.X5,data.c2.complete$data.X6,
data.c2.complete$data.X7,data.c2.complete$data.X8), ncol = 8)
xil.c3.complete =
matrix(c(data.c3.complete$data.X1,data.c3.complete$data.X2,
data.c3.complete$data.X3,data.c3.complete$data.X4,
data.c3.complete$data.X5,data.c3.complete$data.X6,
data.c3.complete$data.X7,data.c3.complete$data.X8), ncol = 8)
xil.c4.complete =
matrix(c(data.c4.complete$data.X1,data.c4.complete$data.X2,
data.c4.complete$data.X3,data.c4.complete$data.X4,
data.c4.complete$data.X5,data.c4.complete$data.X6,
data.c4.complete$data.X7,data.c4.complete$data.X8), ncol = 8)
xil.c5.complete =
matrix(c(data.c5.complete$data.X1,data.c5.complete$data.X2,
data.c5.complete$data.X3,data.c5.complete$data.X4,
data.c5.complete$data.X5,data.c5.complete$data.X6,
data.c5.complete$data.X7,data.c5.complete$data.X8), ncol = 8)
xil.c6.complete =
matrix(c(data.c6.complete$data.X1,data.c6.complete$data.X2,
data.c6.complete$data.X3,data.c6.complete$data.X4,
data.c6.complete$data.X5,data.c6.complete$data.X6,
data.c6.complete$data.X7,data.c6.complete$data.X8), ncol = 8)
x1.complete = sum ((xil.c1.complete-xlc1bar.complete)^2)
x2.complete = sum ((xil.c2.complete-xlc2bar.complete)^2)
x3.complete = sum ((xil.c3.complete-xlc3bar.complete)^2)
x4.complete = sum ((xil.c4.complete-xlc4bar.complete)^2)
x5.complete = sum ((xil.c5.complete-xlc5bar.complete)^2)
x6.complete = sum ((xil.c6.complete-xlc6bar.complete)^2)
SSW.complete.c1 = sum (x1.complete)
SSW.complete.c2 = sum (x1.complete,x2.complete)
SSW.complete.c3 = sum (x1.complete,x2.complete,x3.complete)
SSW.complete.c4 = sum
(x1.complete,x2.complete,x3.complete,x4.complete)
SSW.complete.c5 = sum
(x1.complete,x2.complete,x3.complete,x4.complete,x5.complete)
SSW.complete.c6 = sum
(x1.complete,x2.complete,x3.complete,x4.complete,x5.complete,x6.
complete)
SSB.complete.c1 = SST - SSW.complete.c1
SSB.complete.c2 = SST - SSW.complete.c2
SSB.complete.c3 = SST - SSW.complete.c3
SSB.complete.c4 = SST - SSW.complete.c4
SSB.complete.c5 = SST - SSW.complete.c5
SSB.complete.c6 = SST - SSW.complete.c6
R.square.complete.c1 = SSB.complete.c1/SST
R.square.complete.c2 = SSB.complete.c2/SST

```



```

R.square.complete.c3 = SSB.complete.c3/SST
R.square.complete.c4 = SSB.complete.c4/SST
R.square.complete.c5 = SSB.complete.c5/SST
R.square.complete.c6 = SSB.complete.c6/SST
Hasil.complete = c(SST, SSW.complete.c6,R.square.complete.c6)
# Menghitung sum of square within group (average)
kelompok.average = hasil.kelompok.numerik$average
data.average = data.frame(average, data$X1, data$X2,
data$X3, data$X4, data$X5, data$X6, data$X7, data$X8)
data.average.sort = data.average[order(data.average$average),]
xlc1bar.average = matrix(c(rep(mean(data.c1.average$data.X1),a),
rep(mean(data.c1.average$data.X2),a),
rep(mean(data.c1.average$data.X3),a),
rep(mean(data.c1.average$data.X4),a),
rep(mean(data.c1.average$data.X5),a),
rep(mean(data.c1.average$data.X6),a),
rep(mean(data.c1.average$data.X7),a),
rep(mean(data.c1.average$data.X8),a)), ncol = 8)
xlc2bar.average = matrix(c(rep(mean(data.c2.average$data.X1),b),
rep(mean(data.c2.average$data.X2),b),
rep(mean(data.c2.average$data.X3),b),
rep(mean(data.c2.average$data.X4),b),
rep(mean(data.c2.average$data.X5),b),
rep(mean(data.c2.average$data.X6),b),
rep(mean(data.c2.average$data.X7),b),
rep(mean(data.c2.average$data.X8),b))), ncol = 8)
xlc3bar.average = matrix(c(rep(mean(data.c3.average$data.X1),c),
rep(mean(data.c3.average$data.X2),c),
rep(mean(data.c3.average$data.X3),c),
rep(mean(data.c3.average$data.X4),c),
rep(mean(data.c3.average$data.X5),c),
rep(mean(data.c3.average$data.X6),c),
rep(mean(data.c3.average$data.X7),c),
rep(mean(data.c3.average$data.X8),c))), ncol = 8)
xlc4bar.average = matrix(c(rep(mean(data.c4.average$data.X1),d),
rep(mean(data.c4.average$data.X2),d),
rep(mean(data.c4.average$data.X3),d),
rep(mean(data.c4.average$data.X4),d),
rep(mean(data.c4.average$data.X5),d),
rep(mean(data.c4.average$data.X6),d),
rep(mean(data.c4.average$data.X7),d),
rep(mean(data.c4.average$data.X8),d))), ncol = 8)
xlc5bar.average = matrix(c(rep(mean(data.c5.average$data.X1),e),
rep(mean(data.c5.average$data.X2),e),
rep(mean(data.c5.average$data.X3),e),
rep(mean(data.c5.average$data.X4),e),
rep(mean(data.c5.average$data.X5),e),
rep(mean(data.c5.average$data.X6),e),
rep(mean(data.c5.average$data.X7),e),
rep(mean(data.c5.average$data.X8),e))), ncol = 8)
xlc6bar.average = matrix(c(rep(mean(data.c6.average$data.X1),f),
rep(mean(data.c6.average$data.X2),f),
rep(mean(data.c6.average$data.X3),f),
rep(mean(data.c6.average$data.X4),f),
rep(mean(data.c6.average$data.X5),f),
rep(mean(data.c6.average$data.X6),f),
rep(mean(data.c6.average$data.X7),f),

```

```

rep(mean(data.c6.average$data.X8),f)), ncol = 8)
xil.c1.average =
matrix(c(data.c1.average$data.X1,data.c1.average$data.X2,
data.c1.average$data.X3,data.c1.average$data.X4,
data.c1.average$data.X5,data.c1.average$data.X6,
data.c1.average$data.X7,data.c1.average$data.X8), ncol = 8)
xil.c2.average =
matrix(c(data.c2.average$data.X1,data.c2.average$data.X2,
data.c2.average$data.X3,data.c2.average$data.X4,
data.c2.average$data.X5,data.c2.average$data.X6,
data.c2.average$data.X7,data.c2.average$data.X8), ncol = 8)
xil.c3.average =
matrix(c(data.c3.average$data.X1,data.c3.average$data.X2,
data.c3.average$data.X3,data.c3.average$data.X4,
data.c3.average$data.X5,data.c3.average$data.X6,
data.c3.average$data.X7,data.c3.average$data.X8), ncol = 8)
xil.c4.average =
matrix(c(data.c4.average$data.X1,data.c4.average$data.X2,
data.c4.average$data.X3,data.c4.average$data.X4,
data.c4.average$data.X5,data.c4.average$data.X6,
data.c4.average$data.X7,data.c4.average$data.X8), ncol = 8)
xil.c5.average =
matrix(c(data.c5.average$data.X1,data.c5.average$data.X2,
data.c5.average$data.X3,data.c5.average$data.X4,
data.c5.average$data.X5,data.c5.average$data.X6,
data.c5.average$data.X7,data.c5.average$data.X8), ncol = 8)
xil.c6.average =
matrix(c(data.c6.average$data.X1,data.c6.average$data.X2,
data.c6.average$data.X3,data.c6.average$data.X4,
data.c6.average$data.X5,data.c6.average$data.X6,
data.c6.average$data.X7,data.c6.average$data.X8), ncol = 8)
x1.average = sum ((xil.c1.average-xlc1bar.average)^2)
x2.average = sum ((xil.c2.average-xlc2bar.average)^2)
x3.average = sum ((xil.c3.average-xlc3bar.average)^2)
x4.average = sum ((xil.c4.average-xlc4bar.average)^2)
x5.average = sum ((xil.c5.average-xlc5bar.average)^2)
x6.average = sum ((xil.c6.average-xlc6bar.average)^2)
SSW.average.c1 = sum (x1.average)
SSW.average.c2 = sum (x1.average,x2.average)
SSW.average.c3 = sum (x1.average,x2.average,x3.average)
SSW.average.c4 = sum
(x1.average,x2.average,x3.average,x4.average)
SSW.average.c5 = sum
(x1.average,x2.average,x3.average,x4.average,x5.average)
SSW.average.c6 = sum
(x1.average,x2.average,x3.average,x4.average,x5.average,x6.average)
SSB.average.c1 = SST - SSW.average.c1
SSB.average.c2 = SST - SSW.average.c2
SSB.average.c3 = SST - SSW.average.c3
SSB.average.c4 = SST - SSW.average.c4
SSB.average.c5 = SST - SSW.average.c5
SSB.average.c6 = SST - SSW.average.c6
R.square.average.c1 = SSB.average.c1/SST
R.square.average.c2 = SSB.average.c2/SST
R.square.average.c3 = SSB.average.c3/SST
R.square.average.c4 = SSB.average.c4/SST

```

```

R.square.average.c5 = SSB.average.c5/SST
R.square.average.c6 = SSB.average.c6/SST
Hasil.average = c(SST, SSW.average.c6,R.square.average.c6)
Hasil = rbind(Hasil.single, Hasil.complete, Hasil.average)
print("Hasil Pengelompokan Hirarki")
print(hasil.kelompok.numerik)
print("-----")
print("  Untuk 6 kelompok yang terbentuk  ")
print("Metode      SST      SSW   R-square")
print(Hasil)
}

```

f. Jumlah kelompok yang dibentuk sama dengan 7

```

Analisis.kluster.k7 = function(data){
  k
    = 7
  datanumerik = data.frame(data$X1, data$X2, data$X3, data$X4,
  data$X5, data$X6, data$X7, data$X8)
  d
    = dist(datanumerik, method = "euclidean")
  # Analisis Kluster Hirarki
  fit.sin
    = hclust(d, method = "single")
  fit.com
    = hclust(d, method = "complete")
  fit.ave
    = hclust(d, method = "average")
  # Memotong Dendrogram untuk k Kluster
  single
    = cutree(fit.sin, k=k)
  complete
    = cutree(fit.com, k=k)
  average
    = cutree(fit.ave, k=k)
  hasil.kelompok.numerik =
  data.frame>Nama, single, complete, average)
  # Menghitung sum of square total  $y = (x_{il} - \bar{x}_l)^2$ 
  y = c((data$X1-mean(data$X1))^2, (data$X2-mean(data$X2))^2,
  (data$X3-mean(data$X3))^2, (data$X4-mean(data$X4))^2,
  (data$X5-mean(data$X5))^2, (data$X6-mean(data$X6))^2,
  (data$X7-mean(data$X7))^2, (data$X8-mean(data$X8))^2)
  SST
    = sum (y)
  # Menghitung sum of square within group (single)
  kelompok.single
    = hasil.kelompok.numerik$single
  data.single
    = data.frame(single, data$X1, data$X2,
  data$X3, data$X4, data$X5, data$X6, data$X7, data$X8)
  data.single.sort
    = data.single[order(data.single$single),]
  xlc1bar.single = matrix(c(rep(mean(data.c1.single$data.X1),a),
  rep(mean(data.c1.single$data.X2),a),
  rep(mean(data.c1.single$data.X3),a),
  rep(mean(data.c1.single$data.X4),a),
  rep(mean(data.c1.single$data.X5),a),
  rep(mean(data.c1.single$data.X6),a),
  rep(mean(data.c1.single$data.X7),a),
  rep(mean(data.c1.single$data.X8),a)), ncol = 8)
  xlc2bar.single = matrix(c(rep(mean(data.c2.single$data.X1),b),
  rep(mean(data.c2.single$data.X2),b),
  rep(mean(data.c2.single$data.X3),b),
  rep(mean(data.c2.single$data.X4),b),
  rep(mean(data.c2.single$data.X5),b),
  rep(mean(data.c2.single$data.X6),b),
  rep(mean(data.c2.single$data.X7),b),
  rep(mean(data.c2.single$data.X8),b)), ncol = 8)
  xlc3bar.single = matrix(c(rep(mean(data.c3.single$data.X1),c),

```

```

rep(mean(data.c3.single$data.X2),c),
rep(mean(data.c3.single$data.X3),c),
rep(mean(data.c3.single$data.X4),c),
rep(mean(data.c3.single$data.X5),c),
rep(mean(data.c3.single$data.X6),c),
rep(mean(data.c3.single$data.X7),c),
rep(mean(data.c3.single$data.X8),c)), ncol = 8)
xlc4bar.single = matrix(c(rep(mean(data.c4.single$data.X1),d),
rep(mean(data.c4.single$data.X2),d),
rep(mean(data.c4.single$data.X3),d),
rep(mean(data.c4.single$data.X4),d),
rep(mean(data.c4.single$data.X5),d),
rep(mean(data.c4.single$data.X6),d),
rep(mean(data.c4.single$data.X7),d),
rep(mean(data.c4.single$data.X8),d)), ncol = 8)
xlc5bar.single = matrix(c(rep(mean(data.c5.single$data.X1),e),
rep(mean(data.c5.single$data.X2),e),
rep(mean(data.c5.single$data.X3),e),
rep(mean(data.c5.single$data.X4),e),
rep(mean(data.c5.single$data.X5),e),
rep(mean(data.c5.single$data.X6),e),
rep(mean(data.c5.single$data.X7),e),
rep(mean(data.c5.single$data.X8),e)), ncol = 8)
xlc6bar.single = matrix(c(rep(mean(data.c6.single$data.X1),f),
rep(mean(data.c6.single$data.X2),f),
rep(mean(data.c6.single$data.X3),f),
rep(mean(data.c6.single$data.X4),f),
rep(mean(data.c6.single$data.X5),f),
rep(mean(data.c6.single$data.X6),f),
rep(mean(data.c6.single$data.X7),f),
rep(mean(data.c6.single$data.X8),f)), ncol = 8)
xlc7bar.single = matrix(c(rep(mean(data.c7.single$data.X1),g),
rep(mean(data.c7.single$data.X2),g),
rep(mean(data.c7.single$data.X3),g),
rep(mean(data.c7.single$data.X4),g),
rep(mean(data.c7.single$data.X5),g),
rep(mean(data.c7.single$data.X6),g),
rep(mean(data.c7.single$data.X7),g),
rep(mean(data.c7.single$data.X8),g)), ncol = 8)
xil.c1.single =
matrix(c(data.c1.single$data.X1,data.c1.single$data.X2,
data.c1.single$data.X3,data.c1.single$data.X4,
data.c1.single$data.X5,data.c1.single$data.X6,
data.c1.single$data.X7,data.c1.single$data.X8), ncol = 8)
xil.c2.single =
matrix(c(data.c2.single$data.X1,data.c2.single$data.X2,
data.c2.single$data.X3,data.c2.single$data.X4,
data.c2.single$data.X5,data.c2.single$data.X6,
data.c2.single$data.X7,data.c2.single$data.X8), ncol = 8)
xil.c3.single =
matrix(c(data.c3.single$data.X1,data.c3.single$data.X2,
data.c3.single$data.X3,data.c3.single$data.X4,
data.c3.single$data.X5,data.c3.single$data.X6,
data.c3.single$data.X7,data.c3.single$data.X8), ncol = 8)
xil.c4.single =
matrix(c(data.c4.single$data.X1,data.c4.single$data.X2,
data.c4.single$data.X3,data.c4.single$data.X4,

```

```

data.c4.single$data.X5,data.c4.single$data.X6,
data.c4.single$data.X7,data.c4.single$data.X8), ncol = 8)
xil.c5.single =
matrix(c(data.c5.single$data.X1,data.c5.single$data.X2,
data.c5.single$data.X3,data.c5.single$data.X4,
data.c5.single$data.X5,data.c5.single$data.X6,
data.c5.single$data.X7,data.c5.single$data.X8), ncol = 8)
xil.c6.single =
matrix(c(data.c6.single$data.X1,data.c6.single$data.X2,
data.c6.single$data.X3,data.c6.single$data.X4,
data.c6.single$data.X5,data.c6.single$data.X6,
data.c6.single$data.X7,data.c6.single$data.X8), ncol = 8)
xil.c7.single =
matrix(c(data.c7.single$data.X1,data.c7.single$data.X2,
data.c7.single$data.X3,data.c7.single$data.X4,
data.c7.single$data.X5,data.c7.single$data.X6,
data.c7.single$data.X7,data.c7.single$data.X8), ncol = 8)
x1.single = sum ((xil.c1.single-xlc1bar.single)^2)
x2.single = sum ((xil.c2.single-xlc2bar.single)^2)
x3.single = sum ((xil.c3.single-xlc3bar.single)^2)
x4.single = sum ((xil.c4.single-xlc4bar.single)^2)
x5.single = sum ((xil.c5.single-xlc5bar.single)^2)
x6.single = sum ((xil.c6.single-xlc6bar.single)^2)
x7.single = sum ((xil.c7.single-xlc7bar.single)^2)
SSW.single.c1 = sum (x1.single)
SSW.single.c2 = sum (x1.single,x2.single)
SSW.single.c3 = sum (x1.single,x2.single,x3.single)
SSW.single.c4 = sum (x1.single,x2.single,x3.single,x4.single)
SSW.single.c5 = sum
(x1.single,x2.single,x3.single,x4.single,x5.single)
SSW.single.c6 = sum
(x1.single,x2.single,x3.single,x4.single,x5.single,x6.single)
SSW.single.c7 = sum
(x1.single,x2.single,x3.single,x4.single,x5.single,x6.single,x7.
single)
SSB.single.c1 = SST - SSW.single.c1
SSB.single.c2 = SST - SSW.single.c2
SSB.single.c3 = SST - SSW.single.c3
SSB.single.c4 = SST - SSW.single.c4
SSB.single.c5 = SST - SSW.single.c5
SSB.single.c6 = SST - SSW.single.c6
SSB.single.c7 = SST - SSW.single.c7
R.square.single.c1 = SSB.single.c1/SST
R.square.single.c2 = SSB.single.c2/SST
R.square.single.c3 = SSB.single.c3/SST
R.square.single.c4 = SSB.single.c4/SST
R.square.single.c5 = SSB.single.c5/SST
R.square.single.c6 = SSB.single.c6/SST
R.square.single.c7 = SSB.single.c7/SST
Hasil.single = c(SST, SSW.single.c7, R.square.single.c7)
# Menghitung sum of square within group (complete)
kelompok.complete = hasil.kelompok.numerik$complete
data.complete = data.frame(complete, data$X1, data$X2,
data$X3, data$X4, data$X5, data$X6, data$X7, data$X8)
data.complete.sort =
data.complete[order(data.complete$complete),]

```

```

xlc1bar.complete =
matrix(c(rep(mean(data.c1.complete$data.X1),a),
rep(mean(data.c1.complete$data.X2),a),
rep(mean(data.c1.complete$data.X3),a),
rep(mean(data.c1.complete$data.X4),a),
rep(mean(data.c1.complete$data.X5),a),
rep(mean(data.c1.complete$data.X6),a),
rep(mean(data.c1.complete$data.X7),a),
rep(mean(data.c1.complete$data.X8),a)), ncol = 8)
xlc2bar.complete =
matrix(c(rep(mean(data.c2.complete$data.X1),b),
rep(mean(data.c2.complete$data.X2),b),
rep(mean(data.c2.complete$data.X3),b),
rep(mean(data.c2.complete$data.X4),b),
rep(mean(data.c2.complete$data.X5),b),
rep(mean(data.c2.complete$data.X6),b),
rep(mean(data.c2.complete$data.X7),b),
rep(mean(data.c2.complete$data.X8),b)), ncol = 8)
xlc3bar.complete =
matrix(c(rep(mean(data.c3.complete$data.X1),c),
rep(mean(data.c3.complete$data.X2),c),
rep(mean(data.c3.complete$data.X3),c),
rep(mean(data.c3.complete$data.X4),c),
rep(mean(data.c3.complete$data.X5),c),
rep(mean(data.c3.complete$data.X6),c),
rep(mean(data.c3.complete$data.X7),c),
rep(mean(data.c3.complete$data.X8),c)), ncol = 8)
xlc4bar.complete =
matrix(c(rep(mean(data.c4.complete$data.X1),d),
rep(mean(data.c4.complete$data.X2),d),
rep(mean(data.c4.complete$data.X3),d),
rep(mean(data.c4.complete$data.X4),d),
rep(mean(data.c4.complete$data.X5),d),
rep(mean(data.c4.complete$data.X6),d),
rep(mean(data.c4.complete$data.X7),d),
rep(mean(data.c4.complete$data.X8),d)), ncol = 8)
xlc5bar.complete =
matrix(c(rep(mean(data.c5.complete$data.X1),e),
rep(mean(data.c5.complete$data.X2),e),
rep(mean(data.c5.complete$data.X3),e),
rep(mean(data.c5.complete$data.X4),e),
rep(mean(data.c5.complete$data.X5),e),
rep(mean(data.c5.complete$data.X6),e),
rep(mean(data.c5.complete$data.X7),e),
rep(mean(data.c5.complete$data.X8),e)), ncol = 8)
xlc6bar.complete =
matrix(c(rep(mean(data.c6.complete$data.X1),f),
rep(mean(data.c6.complete$data.X2),f),
rep(mean(data.c6.complete$data.X3),f),
rep(mean(data.c6.complete$data.X4),f),
rep(mean(data.c6.complete$data.X5),f),
rep(mean(data.c6.complete$data.X6),f),
rep(mean(data.c6.complete$data.X7),f),
rep(mean(data.c6.complete$data.X8),f)), ncol = 8)
xlc7bar.complete =
matrix(c(rep(mean(data.c7.complete$data.X1),g),
rep(mean(data.c7.complete$data.X2),g),

```

```

rep(mean(data.c7.complete$data.X3),g),
rep(mean(data.c7.complete$data.X4),g),
rep(mean(data.c7.complete$data.X5),g),
rep(mean(data.c7.complete$data.X6),g),
rep(mean(data.c7.complete$data.X7),g),
rep(mean(data.c7.complete$data.X8),g)), ncol = 8)
xil.c1.complete =
matrix(c(data.c1.complete$data.X1,data.c1.complete$data.X2,
data.c1.complete$data.X3,data.c1.complete$data.X4,
data.c1.complete$data.X5,data.c1.complete$data.X6,
data.c1.complete$data.X7,data.c1.complete$data.X8), ncol = 8)
xil.c2.complete =
matrix(c(data.c2.complete$data.X1,data.c2.complete$data.X2,
data.c2.complete$data.X3,data.c2.complete$data.X4,
data.c2.complete$data.X5,data.c2.complete$data.X6,
data.c2.complete$data.X7,data.c2.complete$data.X8), ncol = 8)
xil.c3.complete =
matrix(c(data.c3.complete$data.X1,data.c3.complete$data.X2,
data.c3.complete$data.X3,data.c3.complete$data.X4,
data.c3.complete$data.X5,data.c3.complete$data.X6,
data.c3.complete$data.X7,data.c3.complete$data.X8), ncol = 8)
xil.c4.complete =
matrix(c(data.c4.complete$data.X1,data.c4.complete$data.X2,
data.c4.complete$data.X3,data.c4.complete$data.X4,
data.c4.complete$data.X5,data.c4.complete$data.X6,
data.c4.complete$data.X7,data.c4.complete$data.X8), ncol = 8)
xil.c5.complete =
matrix(c(data.c5.complete$data.X1,data.c5.complete$data.X2,
data.c5.complete$data.X3,data.c5.complete$data.X4,
data.c5.complete$data.X5,data.c5.complete$data.X6,
data.c5.complete$data.X7,data.c5.complete$data.X8), ncol = 8)
xil.c6.complete =
matrix(c(data.c6.complete$data.X1,data.c6.complete$data.X2,
data.c6.complete$data.X3,data.c6.complete$data.X4,
data.c6.complete$data.X5,data.c6.complete$data.X6,
data.c6.complete$data.X7,data.c6.complete$data.X8), ncol = 8)
xil.c7.complete =
matrix(c(data.c7.complete$data.X1,data.c7.complete$data.X2,
data.c7.complete$data.X3,data.c7.complete$data.X4,
data.c7.complete$data.X5,data.c7.complete$data.X6,
data.c7.complete$data.X7,data.c7.complete$data.X8), ncol = 8)
x1.complete = sum ((xil.c1.complete-xlc1bar.complete)^2)
x2.complete = sum ((xil.c2.complete-xlc2bar.complete)^2)
x3.complete = sum ((xil.c3.complete-xlc3bar.complete)^2)
x4.complete = sum ((xil.c4.complete-xlc4bar.complete)^2)
x5.complete = sum ((xil.c5.complete-xlc5bar.complete)^2)
x6.complete = sum ((xil.c6.complete-xlc6bar.complete)^2)
x7.complete = sum ((xil.c7.complete-xlc7bar.complete)^2)
SSW.complete.c1 = sum (x1.complete)
SSW.complete.c2 = sum (x1.complete,x2.complete)
SSW.complete.c3 = sum (x1.complete,x2.complete,x3.complete)
SSW.complete.c4 = sum
(x1.complete,x2.complete,x3.complete,x4.complete)
SSW.complete.c5 = sum
(x1.complete,x2.complete,x3.complete,x4.complete,x5.complete)

```

```

SSW.complete.c6 = sum
(x1.complete,x2.complete,x3.complete,x4.complete,x5.complete,x6.
complete)
SSW.complete.c7 = sum
(x1.complete,x2.complete,x3.complete,x4.complete,x5.complete,x6.
complete,x7.complete)
SSB.complete.c1 = SST - SSW.complete.c1
SSB.complete.c2 = SST - SSW.complete.c2
SSB.complete.c3 = SST - SSW.complete.c3
SSB.complete.c4 = SST - SSW.complete.c4
SSB.complete.c5 = SST - SSW.complete.c5
SSB.complete.c6 = SST - SSW.complete.c6
SSB.complete.c7 = SST - SSW.complete.c7
R.square.complete.c1 = SSB.complete.c1/SST
R.square.complete.c2 = SSB.complete.c2/SST
R.square.complete.c3 = SSB.complete.c3/SST
R.square.complete.c4 = SSB.complete.c4/SST
R.square.complete.c5 = SSB.complete.c5/SST
R.square.complete.c6 = SSB.complete.c6/SST
R.square.complete.c7 = SSB.complete.c7/SST
Hasil.complete = c(SST, SSW.complete.c7, R.square.complete.c7)
# Menghitung sum of square within group (average)
kelompok.average = hasil.kelompok.numerik$average
data.average = data.frame(average, data$X1, data$X2,
data$X3, data$X4, data$X5, data$X6, data$X7, data$X8)
data.average.sort = data.average[order(data.average$average),]
xlc1bar.average = matrix(c(rep(mean(data.c1.average$data.X1),a),
rep(mean(data.c1.average$data.X2),a),
rep(mean(data.c1.average$data.X3),a),
rep(mean(data.c1.average$data.X4),a),
rep(mean(data.c1.average$data.X5),a),
rep(mean(data.c1.average$data.X6),a),
rep(mean(data.c1.average$data.X7),a),
rep(mean(data.c1.average$data.X8),a)), ncol = 8)
xlc2bar.average = matrix(c(rep(mean(data.c2.average$data.X1),b),
rep(mean(data.c2.average$data.X2),b),
rep(mean(data.c2.average$data.X3),b),
rep(mean(data.c2.average$data.X4),b),
rep(mean(data.c2.average$data.X5),b),
rep(mean(data.c2.average$data.X6),b),
rep(mean(data.c2.average$data.X7),b),
rep(mean(data.c2.average$data.X8),b)), ncol = 8)
xlc3bar.average = matrix(c(rep(mean(data.c3.average$data.X1),c),
rep(mean(data.c3.average$data.X2),c),
rep(mean(data.c3.average$data.X3),c),
rep(mean(data.c3.average$data.X4),c),
rep(mean(data.c3.average$data.X5),c),
rep(mean(data.c3.average$data.X6),c),
rep(mean(data.c3.average$data.X7),c),
rep(mean(data.c3.average$data.X8),c)), ncol = 8)
xlc4bar.average = matrix(c(rep(mean(data.c4.average$data.X1),d),
rep(mean(data.c4.average$data.X2),d),
rep(mean(data.c4.average$data.X3),d),
rep(mean(data.c4.average$data.X4),d),
rep(mean(data.c4.average$data.X5),d),
rep(mean(data.c4.average$data.X6),d),
rep(mean(data.c4.average$data.X7),d),

```



```

rep(mean(data.c4.average$data.X8),d)), ncol = 8)
xlc5bar.average = matrix(c(rep(mean(data.c5.average$data.X1),e),
rep(mean(data.c5.average$data.X2),e),
rep(mean(data.c5.average$data.X3),e),
rep(mean(data.c5.average$data.X4),e),
rep(mean(data.c5.average$data.X5),e),
rep(mean(data.c5.average$data.X6),e),
rep(mean(data.c5.average$data.X7),e),
rep(mean(data.c5.average$data.X8),e))), ncol = 8)
xlc6bar.average = matrix(c(rep(mean(data.c6.average$data.X1),f),
rep(mean(data.c6.average$data.X2),f),
rep(mean(data.c6.average$data.X3),f),
rep(mean(data.c6.average$data.X4),f),
rep(mean(data.c6.average$data.X5),f),
rep(mean(data.c6.average$data.X6),f),
rep(mean(data.c6.average$data.X7),f),
rep(mean(data.c6.average$data.X8),f))), ncol = 8)
xlc7bar.average = matrix(c(rep(mean(data.c7.average$data.X1),g),
rep(mean(data.c7.average$data.X2),g),
rep(mean(data.c7.average$data.X3),g),
rep(mean(data.c7.average$data.X4),g),
rep(mean(data.c7.average$data.X5),g),
rep(mean(data.c7.average$data.X6),g),
rep(mean(data.c7.average$data.X7),g),
rep(mean(data.c7.average$data.X8),g))), ncol = 8)
xil.c1.average =
matrix(c(data.c1.average$data.X1,data.c1.average$data.X2,
data.c1.average$data.X3,data.c1.average$data.X4,
data.c1.average$data.X5,data.c1.average$data.X6,
data.c1.average$data.X7,data.c1.average$data.X8), ncol = 8)
xil.c2.average =
matrix(c(data.c2.average$data.X1,data.c2.average$data.X2,
data.c2.average$data.X3,data.c2.average$data.X4,
data.c2.average$data.X5,data.c2.average$data.X6,
data.c2.average$data.X7,data.c2.average$data.X8), ncol = 8)
xil.c3.average =
matrix(c(data.c3.average$data.X1,data.c3.average$data.X2,
data.c3.average$data.X3,data.c3.average$data.X4,
data.c3.average$data.X5,data.c3.average$data.X6,
data.c3.average$data.X7,data.c3.average$data.X8), ncol = 8)
xil.c4.average =
matrix(c(data.c4.average$data.X1,data.c4.average$data.X2,
data.c4.average$data.X3,data.c4.average$data.X4,
data.c4.average$data.X5,data.c4.average$data.X6,
data.c4.average$data.X7,data.c4.average$data.X8), ncol = 8)
xil.c5.average =
matrix(c(data.c5.average$data.X1,data.c5.average$data.X2,
data.c5.average$data.X3,data.c5.average$data.X4,
data.c5.average$data.X5,data.c5.average$data.X6,
data.c5.average$data.X7,data.c5.average$data.X8), ncol = 8)
xil.c6.average =
matrix(c(data.c6.average$data.X1,data.c6.average$data.X2,
data.c6.average$data.X3,data.c6.average$data.X4,
data.c6.average$data.X5,data.c6.average$data.X6,
data.c6.average$data.X7,data.c6.average$data.X8), ncol = 8)
xil.c7.average =
matrix(c(data.c7.average$data.X1,data.c7.average$data.X2,

```

```

data.c7.average$data.X3,data.c7.average$data.X4,
data.c7.average$data.X5,data.c7.average$data.X6,
data.c7.average$data.X7,data.c7.average$data.X8), ncol = 8)
x1.average = sum ((xil.c1.average-xlc1bar.average)^2)
x2.average = sum ((xil.c2.average-xlc2bar.average)^2)
x3.average = sum ((xil.c3.average-xlc3bar.average)^2)
x4.average = sum ((xil.c4.average-xlc4bar.average)^2)
x5.average = sum ((xil.c5.average-xlc5bar.average)^2)
x6.average = sum ((xil.c6.average-xlc6bar.average)^2)
x7.average = sum ((xil.c7.average-xlc7bar.average)^2)
SSW.average.c1 = sum (x1.average)
SSW.average.c2 = sum (x1.average,x2.average)
SSW.average.c3 = sum (x1.average,x2.average,x3.average)
SSW.average.c4 = sum
(x1.average,x2.average,x3.average,x4.average)
SSW.average.c5 = sum
(x1.average,x2.average,x3.average,x4.average,x5.average)
SSW.average.c6 = sum
(x1.average,x2.average,x3.average,x4.average,x5.average,x6.average)
SSW.average.c7 = sum
(x1.average,x2.average,x3.average,x4.average,x5.average,x6.average,x7.average)

SSB.average.c1 = SST - SSW.average.c1
SSB.average.c2 = SST - SSW.average.c2
SSB.average.c3 = SST - SSW.average.c3
SSB.average.c4 = SST - SSW.average.c4
SSB.average.c5 = SST - SSW.average.c5
SSB.average.c6 = SST - SSW.average.c6
SSB.average.c7 = SST - SSW.average.c7
R.square.average.c1 = SSB.average.c1/SST
R.square.average.c2 = SSB.average.c2/SST
R.square.average.c3 = SSB.average.c3/SST
R.square.average.c4 = SSB.average.c4/SST
R.square.average.c5 = SSB.average.c5/SST
R.square.average.c6 = SSB.average.c6/SST
R.square.average.c7 = SSB.average.c7/SST
Hasil.average = c(SST, SSW.average.c7, R.square.average.c7)
Hasil = rbind(Hasil.single, Hasil.complete, Hasil.average)
print("Hasil Pengelompokan Hirarki")
print(hasil.kelompok.numerik)
print("-----")
print(" Untuk 7 kelompok yang terbentuk ")
print("Metode SST SSW R-square")
print(Hasil)
}

```

g. Jumlah kelompok yang dibentuk sama dengan 8

```

Analisis.kluster.k8 = function(data){
k
= 8
datanumerik = data.frame(data$X1, data$X2, data$X3, data$X4,
data$X5, data$X6, data$X7, data$X8)
d
= dist(datanumerik, method = "euclidean")
# Analisis Kluster Hirarki
fit.sin
= hclust(d, method = "single")

```

```

fit.com      = hclust(d, method = "complete")
fit.ave      = hclust(d, method = "average")
# Memotong Dendrogram untuk k Kluster
single      = cutree(fit.sin, k=k)
complete    = cutree(fit.com, k=k)
average     = cutree(fit.ave, k=k)
hasil.kelompok.numerik = data.frame>Nama, single, complete,
average)
# Menghitung sum of square total  $y = (x_{il} - \bar{x}_l)^2$ 
y = c((data$X1 - mean(data$X1))^2, (data$X2 - mean(data$X2))^2,
(data$X3 - mean(data$X3))^2, (data$X4 - mean(data$X4))^2,
(data$X5 - mean(data$X5))^2, (data$X6 - mean(data$X6))^2,
(data$X7 - mean(data$X7))^2, (data$X8 - mean(data$X8))^2)
SST      = sum(y)
# Menghitung sum of square within group (single)
kelompok.single = hasil.kelompok.numerik$single
data.single      = data.frame(single, data$X1, data$X2,
data$X3, data$X4, data$X5, data$X6, data$X7, data$X8)
data.single.sort  = data.single[order(data.single$single),]
xlc1bar.single    = matrix(c(rep(mean(data.c1.single$data.X1), a),
rep(mean(data.c1.single$data.X2), a),
rep(mean(data.c1.single$data.X3), a),
rep(mean(data.c1.single$data.X4), a),
rep(mean(data.c1.single$data.X5), a),
rep(mean(data.c1.single$data.X6), a),
rep(mean(data.c1.single$data.X7), a),
rep(mean(data.c1.single$data.X8), a)), ncol = 8)
xlc2bar.single    = matrix(c(rep(mean(data.c2.single$data.X1), b),
rep(mean(data.c2.single$data.X2), b),
rep(mean(data.c2.single$data.X3), b),
rep(mean(data.c2.single$data.X4), b),
rep(mean(data.c2.single$data.X5), b),
rep(mean(data.c2.single$data.X6), b),
rep(mean(data.c2.single$data.X7), b),
rep(mean(data.c2.single$data.X8), b)), ncol = 8)
xlc3bar.single    = matrix(c(rep(mean(data.c3.single$data.X1), c),
rep(mean(data.c3.single$data.X2), c),
rep(mean(data.c3.single$data.X3), c),
rep(mean(data.c3.single$data.X4), c),
rep(mean(data.c3.single$data.X5), c),
rep(mean(data.c3.single$data.X6), c),
rep(mean(data.c3.single$data.X7), c),
rep(mean(data.c3.single$data.X8), c)), ncol = 8)
xlc4bar.single    = matrix(c(rep(mean(data.c4.single$data.X1), d),
rep(mean(data.c4.single$data.X2), d),
rep(mean(data.c4.single$data.X3), d),
rep(mean(data.c4.single$data.X4), d),
rep(mean(data.c4.single$data.X5), d),
rep(mean(data.c4.single$data.X6), d),
rep(mean(data.c4.single$data.X7), d),
rep(mean(data.c4.single$data.X8), d)), ncol = 8)
xlc5bar.single    = matrix(c(rep(mean(data.c5.single$data.X1), e),
rep(mean(data.c5.single$data.X2), e),
rep(mean(data.c5.single$data.X3), e),
rep(mean(data.c5.single$data.X4), e),
rep(mean(data.c5.single$data.X5), e),
rep(mean(data.c5.single$data.X6), e),

```

```

rep(mean(data.c5.single$data.X7),e),
rep(mean(data.c5.single$data.X8),e)), ncol = 8)
xlc6bar.single = matrix(c(rep(mean(data.c6.single$data.X1),f),
rep(mean(data.c6.single$data.X2),f),
rep(mean(data.c6.single$data.X3),f),
rep(mean(data.c6.single$data.X4),f),
rep(mean(data.c6.single$data.X5),f),
rep(mean(data.c6.single$data.X6),f),
rep(mean(data.c6.single$data.X7),f),
rep(mean(data.c6.single$data.X8),f)), ncol = 8)
xlc7bar.single = matrix(c(rep(mean(data.c7.single$data.X1),g),
rep(mean(data.c7.single$data.X2),g),
rep(mean(data.c7.single$data.X3),g),
rep(mean(data.c7.single$data.X4),g),
rep(mean(data.c7.single$data.X5),g),
rep(mean(data.c7.single$data.X6),g),
rep(mean(data.c7.single$data.X7),g),
rep(mean(data.c7.single$data.X8),g)), ncol = 8)
xlc8bar.single = matrix(c(rep(mean(data.c8.single$data.X1),h),
rep(mean(data.c8.single$data.X2),h),
rep(mean(data.c8.single$data.X3),h),
rep(mean(data.c8.single$data.X4),h),
rep(mean(data.c8.single$data.X5),h),
rep(mean(data.c8.single$data.X6),h),
rep(mean(data.c8.single$data.X7),h),
rep(mean(data.c8.single$data.X8),h)), ncol = 8)
xil.c1.single =
matrix(c(data.c1.single$data.X1,data.c1.single$data.X2,
data.c1.single$data.X3,data.c1.single$data.X4,
data.c1.single$data.X5,data.c1.single$data.X6,
data.c1.single$data.X7,data.c1.single$data.X8), ncol = 8)
xil.c2.single =
matrix(c(data.c2.single$data.X1,data.c2.single$data.X2,
data.c2.single$data.X3,data.c2.single$data.X4,
data.c2.single$data.X5,data.c2.single$data.X6,
data.c2.single$data.X7,data.c2.single$data.X8), ncol = 8)
xil.c3.single =
matrix(c(data.c3.single$data.X1,data.c3.single$data.X2,
data.c3.single$data.X3,data.c3.single$data.X4,
data.c3.single$data.X5,data.c3.single$data.X6,
data.c3.single$data.X7,data.c3.single$data.X8), ncol = 8)
xil.c4.single =
matrix(c(data.c4.single$data.X1,data.c4.single$data.X2,
data.c4.single$data.X3,data.c4.single$data.X4,
data.c4.single$data.X5,data.c4.single$data.X6,
data.c4.single$data.X7,data.c4.single$data.X8), ncol = 8)
xil.c5.single =
matrix(c(data.c5.single$data.X1,data.c5.single$data.X2,
data.c5.single$data.X3,data.c5.single$data.X4,
data.c5.single$data.X5,data.c5.single$data.X6,
data.c5.single$data.X7,data.c5.single$data.X8), ncol = 8)
xil.c6.single =
matrix(c(data.c6.single$data.X1,data.c6.single$data.X2,
data.c6.single$data.X3,data.c6.single$data.X4,
data.c6.single$data.X5,data.c6.single$data.X6,
data.c6.single$data.X7,data.c6.single$data.X8), ncol = 8)

```

```

x1l.c7.single =
matrix(c(data.c7.single$data.X1,data.c7.single$data.X2,
data.c7.single$data.X3,data.c7.single$data.X4,
data.c7.single$data.X5,data.c7.single$data.X6,
data.c7.single$data.X7,data.c7.single$data.X8), ncol = 8)
x1l.c8.single =
matrix(c(data.c8.single$data.X1,data.c8.single$data.X2,
data.c8.single$data.X3,data.c8.single$data.X4,
data.c8.single$data.X5,data.c8.single$data.X6,
data.c8.single$data.X7,data.c8.single$data.X8), ncol = 8)
x1.single = sum ((x1l.c1.single-x1c1bar.single)^2)
x2.single = sum ((x1l.c2.single-x1c2bar.single)^2)
x3.single = sum ((x1l.c3.single-x1c3bar.single)^2)
x4.single = sum ((x1l.c4.single-x1c4bar.single)^2)
x5.single = sum ((x1l.c5.single-x1c5bar.single)^2)
x6.single = sum ((x1l.c6.single-x1c6bar.single)^2)
x7.single = sum ((x1l.c7.single-x1c7bar.single)^2)
x8.single = sum ((x1l.c8.single-x1c8bar.single)^2)
SSW.single.c1 = sum (x1.single)
SSW.single.c2 = sum (x1.single,x2.single)
SSW.single.c3 = sum (x1.single,x2.single,x3.single)
SSW.single.c4 = sum (x1.single,x2.single,x3.single,x4.single)
SSW.single.c5 = sum
(x1.single,x2.single,x3.single,x4.single,x5.single)
SSW.single.c6 = sum
(x1.single,x2.single,x3.single,x4.single,x5.single,x6.single)
SSW.single.c7 = sum
(x1.single,x2.single,x3.single,x4.single,x5.single,x6.single,x7.
single)
SSW.single.c8 = sum
(x1.single,x2.single,x3.single,x4.single,x5.single,x6.single,x7.
single,x8.single)
SSB.single.c1 = SST - SSW.single.c1
SSB.single.c2 = SST - SSW.single.c2
SSB.single.c3 = SST - SSW.single.c3
SSB.single.c4 = SST - SSW.single.c4
SSB.single.c5 = SST - SSW.single.c5
SSB.single.c6 = SST - SSW.single.c6
SSB.single.c7 = SST - SSW.single.c7
SSB.single.c8 = SST - SSW.single.c8
R.square.single.c1 = SSB.single.c1/SST
R.square.single.c2 = SSB.single.c2/SST
R.square.single.c3 = SSB.single.c3/SST
R.square.single.c4 = SSB.single.c4/SST
R.square.single.c5 = SSB.single.c5/SST
R.square.single.c6 = SSB.single.c6/SST
R.square.single.c7 = SSB.single.c7/SST
R.square.single.c8 = SSB.single.c8/SST
Hasil.single = c(SST, SSW.single.c8, R.square.single.c8)
# Menghitung sum of square within group (complete)
kelompok.complete = hasil.kelompok.numerik$complete
data.complete = data.frame(complete, data$X1, data$X2,
data$X3, data$X4, data$X5, data$X6, data$X7, data$X8)
data.complete.sort =
data.complete[order(data.complete$complete),]
x1c1bar.complete =
matrix(c(rep(mean(data.c1.complete$data.X1),a),

```

```

rep(mean(data.c1.complete$data.X2),a),
rep(mean(data.c1.complete$data.X3),a),
rep(mean(data.c1.complete$data.X4),a),
rep(mean(data.c1.complete$data.X5),a),
rep(mean(data.c1.complete$data.X6),a),
rep(mean(data.c1.complete$data.X7),a),
rep(mean(data.c1.complete$data.X8),a)), ncol = 8)
xlc2bar.complete =
matrix(c(rep(mean(data.c2.complete$data.X1),b),
rep(mean(data.c2.complete$data.X2),b),
rep(mean(data.c2.complete$data.X3),b),
rep(mean(data.c2.complete$data.X4),b),
rep(mean(data.c2.complete$data.X5),b),
rep(mean(data.c2.complete$data.X6),b),
rep(mean(data.c2.complete$data.X7),b),
rep(mean(data.c2.complete$data.X8),b))), ncol = 8)
xlc3bar.complete =
matrix(c(rep(mean(data.c3.complete$data.X1),c),
rep(mean(data.c3.complete$data.X2),c),
rep(mean(data.c3.complete$data.X3),c),
rep(mean(data.c3.complete$data.X4),c),
rep(mean(data.c3.complete$data.X5),c),
rep(mean(data.c3.complete$data.X6),c),
rep(mean(data.c3.complete$data.X7),c),
rep(mean(data.c3.complete$data.X8),c))), ncol = 8)
xlc4bar.complete = matrix(c(
rep(mean(data.c4.complete$data.X1),d),
rep(mean(data.c4.complete$data.X2),d),
rep(mean(data.c4.complete$data.X3),d),
rep(mean(data.c4.complete$data.X4),d),
rep(mean(data.c4.complete$data.X5),d),
rep(mean(data.c4.complete$data.X6),d),
rep(mean(data.c4.complete$data.X7),d),
rep(mean(data.c4.complete$data.X8),d))), ncol = 8)
xlc5bar.complete =
matrix(c(rep(mean(data.c5.complete$data.X1),e),
rep(mean(data.c5.complete$data.X2),e),
rep(mean(data.c5.complete$data.X3),e),
rep(mean(data.c5.complete$data.X4),e),
rep(mean(data.c5.complete$data.X5),e),
rep(mean(data.c5.complete$data.X6),e),
rep(mean(data.c5.complete$data.X7),e),
rep(mean(data.c5.complete$data.X8),e))), ncol = 8)
xlc6bar.complete =
matrix(c(rep(mean(data.c6.complete$data.X1),f),
rep(mean(data.c6.complete$data.X2),f),
rep(mean(data.c6.complete$data.X3),f),
rep(mean(data.c6.complete$data.X4),f),
rep(mean(data.c6.complete$data.X5),f),
rep(mean(data.c6.complete$data.X6),f),
rep(mean(data.c6.complete$data.X7),f),
rep(mean(data.c6.complete$data.X8),f))), ncol = 8)
xlc7bar.complete =
matrix(c(rep(mean(data.c7.complete$data.X1),g),
rep(mean(data.c7.complete$data.X2),g),
rep(mean(data.c7.complete$data.X3),g),
rep(mean(data.c7.complete$data.X4),g),

```

```

rep(mean(data.c7.complete$data.X5),g),
rep(mean(data.c7.complete$data.X6),g),
rep(mean(data.c7.complete$data.X7),g),
rep(mean(data.c7.complete$data.X8),g)), ncol = 8)
xlc8bar.complete =
matrix(c(rep(mean(data.c8.complete$data.X1),h),
rep(mean(data.c8.complete$data.X2),h),
rep(mean(data.c8.complete$data.X3),h),
rep(mean(data.c8.complete$data.X4),h),
rep(mean(data.c8.complete$data.X5),h),
rep(mean(data.c8.complete$data.X6),h),
rep(mean(data.c8.complete$data.X7),h),
rep(mean(data.c8.complete$data.X8),h)), ncol = 8)
xil.c1.complete =
matrix(c(data.c1.complete$data.X1,data.c1.complete$data.X2,
data.c1.complete$data.X3,data.c1.complete$data.X4,
data.c1.complete$data.X5,data.c1.complete$data.X6,
data.c1.complete$data.X7,data.c1.complete$data.X8), ncol = 8)
xil.c2.complete =
matrix(c(data.c2.complete$data.X1,data.c2.complete$data.X2,
data.c2.complete$data.X3,data.c2.complete$data.X4,
data.c2.complete$data.X5,data.c2.complete$data.X6,
data.c2.complete$data.X7,data.c2.complete$data.X8), ncol = 8)
xil.c3.complete =
matrix(c(data.c3.complete$data.X1,data.c3.complete$data.X2,
data.c3.complete$data.X3,data.c3.complete$data.X4,
data.c3.complete$data.X5,data.c3.complete$data.X6,
data.c3.complete$data.X7,data.c3.complete$data.X8), ncol = 8)
xil.c4.complete =
matrix(c(data.c4.complete$data.X1,data.c4.complete$data.X2,
data.c4.complete$data.X3,data.c4.complete$data.X4,
data.c4.complete$data.X5,data.c4.complete$data.X6,
data.c4.complete$data.X7,data.c4.complete$data.X8), ncol = 8)
xil.c5.complete =
matrix(c(data.c5.complete$data.X1,data.c5.complete$data.X2,
data.c5.complete$data.X3,data.c5.complete$data.X4,
data.c5.complete$data.X5,data.c5.complete$data.X6,
data.c5.complete$data.X7,data.c5.complete$data.X8), ncol = 8)
xil.c6.complete =
matrix(c(data.c6.complete$data.X1,data.c6.complete$data.X2,
data.c6.complete$data.X3,data.c6.complete$data.X4,
data.c6.complete$data.X5,data.c6.complete$data.X6,
data.c6.complete$data.X7,data.c6.complete$data.X8), ncol = 8)
xil.c7.complete =
matrix(c(data.c7.complete$data.X1,data.c7.complete$data.X2,
data.c7.complete$data.X3,data.c7.complete$data.X4,
data.c7.complete$data.X5,data.c7.complete$data.X6,
data.c7.complete$data.X7,data.c7.complete$data.X8), ncol = 8)
xil.c8.complete =
matrix(c(data.c8.complete$data.X1,data.c8.complete$data.X2,
data.c8.complete$data.X3,data.c8.complete$data.X4,
data.c8.complete$data.X5,data.c8.complete$data.X6,
data.c8.complete$data.X7,data.c8.complete$data.X8), ncol = 8)
x1.complete = sum ((xil.c1.complete-xlc1bar.complete)^2)
x2.complete = sum ((xil.c2.complete-xlc2bar.complete)^2)
x3.complete = sum ((xil.c3.complete-xlc3bar.complete)^2)
x4.complete = sum ((xil.c4.complete-xlc4bar.complete)^2)

```

```

x5.complete = sum ((xil.c5.complete-xlc5bar.complete)^2)
x6.complete = sum ((xil.c6.complete-xlc6bar.complete)^2)
x7.complete = sum ((xil.c7.complete-xlc7bar.complete)^2)
x8.complete = sum ((xil.c8.complete-xlc8bar.complete)^2)
SSW.complete.c1 = sum (x1.complete)
SSW.complete.c2 = sum (x1.complete,x2.complete)
SSW.complete.c3 = sum (x1.complete,x2.complete,x3.complete)
SSW.complete.c4 = sum
(x1.complete,x2.complete,x3.complete,x4.complete)
SSW.complete.c5 = sum
(x1.complete,x2.complete,x3.complete,x4.complete,x5.complete)
SSW.complete.c6 = sum
(x1.complete,x2.complete,x3.complete,x4.complete,x5.complete,x6.
complete)
SSW.complete.c7 = sum
(x1.complete,x2.complete,x3.complete,x4.complete,x5.complete,x6.
complete,x7.complete)
SSW.complete.c8 = sum
(x1.complete,x2.complete,x3.complete,x4.complete,x5.complete,x6.
complete,x7.complete,x8.complete)
SSB.complete.c1 = SST - SSW.complete.c1
SSB.complete.c2 = SST - SSW.complete.c2
SSB.complete.c3 = SST - SSW.complete.c3
SSB.complete.c4 = SST - SSW.complete.c4
SSB.complete.c5 = SST - SSW.complete.c5
SSB.complete.c6 = SST - SSW.complete.c6
SSB.complete.c7 = SST - SSW.complete.c7
SSB.complete.c8 = SST - SSW.complete.c8
R.square.complete.c1 = SSB.complete.c1/SST
R.square.complete.c2 = SSB.complete.c2/SST
R.square.complete.c3 = SSB.complete.c3/SST
R.square.complete.c4 = SSB.complete.c4/SST
R.square.complete.c5 = SSB.complete.c5/SST
R.square.complete.c6 = SSB.complete.c6/SST
R.square.complete.c7 = SSB.complete.c7/SST
R.square.complete.c8 = SSB.complete.c8/SST
R.square.complete.c8 = SSB.complete.c8/SST
Hasil.complete = c(SST, SSW.complete.c8, R.square.complete.c8)
# Menghitung sum of square within group (average)
kelompok.average = hasil.kelompok.numerik$average
data.average = data.frame(average, data$X1, data$X2,
data$X3, data$X4, data$X5, data$X6, data$X7, data$X8)
data.average.sort = data.average[order(data.average$average),]
xlc1bar.average = matrix(c(rep(mean(data.c1.average$data.X1),a),
rep(mean(data.c1.average$data.X2),a),
rep(mean(data.c1.average$data.X3),a),
rep(mean(data.c1.average$data.X4),a),
rep(mean(data.c1.average$data.X5),a),
rep(mean(data.c1.average$data.X6),a),
rep(mean(data.c1.average$data.X7),a),
rep(mean(data.c1.average$data.X8),a)), ncol = 8)
xlc2bar.average = matrix(c(rep(mean(data.c2.average$data.X1),b),
rep(mean(data.c2.average$data.X2),b),
rep(mean(data.c2.average$data.X3),b),
rep(mean(data.c2.average$data.X4),b),
rep(mean(data.c2.average$data.X5),b),
rep(mean(data.c2.average$data.X6),b),

```



```

rep(mean(data.c2.average$data.X7),b),
rep(mean(data.c2.average$data.X8),b)), ncol = 8)
xlc3bar.average = matrix(c(rep(mean(data.c3.average$data.X1),c),
rep(mean(data.c3.average$data.X2),c),
rep(mean(data.c3.average$data.X3),c),
rep(mean(data.c3.average$data.X4),c),
rep(mean(data.c3.average$data.X5),c),
rep(mean(data.c3.average$data.X6),c),
rep(mean(data.c3.average$data.X7),c),
rep(mean(data.c3.average$data.X8),c)), ncol = 8)
xlc4bar.average = matrix(c(rep(mean(data.c4.average$data.X1),d),
rep(mean(data.c4.average$data.X2),d),
rep(mean(data.c4.average$data.X3),d),
rep(mean(data.c4.average$data.X4),d),
rep(mean(data.c4.average$data.X5),d),
rep(mean(data.c4.average$data.X6),d),
rep(mean(data.c4.average$data.X7),d),
rep(mean(data.c4.average$data.X8),d)), ncol = 8)
xlc5bar.average = matrix(c(rep(mean(data.c5.average$data.X1),e),
rep(mean(data.c5.average$data.X2),e),
rep(mean(data.c5.average$data.X3),e),
rep(mean(data.c5.average$data.X4),e),
rep(mean(data.c5.average$data.X5),e),
rep(mean(data.c5.average$data.X6),e),
rep(mean(data.c5.average$data.X7),e),
rep(mean(data.c5.average$data.X8),e)), ncol = 8)
xlc6bar.average = matrix(c(rep(mean(data.c6.average$data.X1),f),
rep(mean(data.c6.average$data.X2),f),
rep(mean(data.c6.average$data.X3),f),
rep(mean(data.c6.average$data.X4),f),
rep(mean(data.c6.average$data.X5),f),
rep(mean(data.c6.average$data.X6),f),
rep(mean(data.c6.average$data.X7),f),
rep(mean(data.c6.average$data.X8),f)), ncol = 8)
xlc7bar.average = matrix(c(rep(mean(data.c7.average$data.X1),g),
rep(mean(data.c7.average$data.X2),g),
rep(mean(data.c7.average$data.X3),g),
rep(mean(data.c7.average$data.X4),g),
rep(mean(data.c7.average$data.X5),g),
rep(mean(data.c7.average$data.X6),g),
rep(mean(data.c7.average$data.X7),g),
rep(mean(data.c7.average$data.X8),g)), ncol = 8)
xlc8bar.average = matrix(c(rep(mean(data.c8.average$data.X1),h),
rep(mean(data.c8.average$data.X2),h),
rep(mean(data.c8.average$data.X3),h),
rep(mean(data.c8.average$data.X4),h),
rep(mean(data.c8.average$data.X5),h),
rep(mean(data.c8.average$data.X6),h),
rep(mean(data.c8.average$data.X7),h),
rep(mean(data.c8.average$data.X8),h)), ncol = 8)
xil.c1.average =
matrix(c(data.c1.average$data.X1,data.c1.average$data.X2,
data.c1.average$data.X3,data.c1.average$data.X4,
data.c1.average$data.X5,data.c1.average$data.X6,
data.c1.average$data.X7,data.c1.average$data.X8), ncol = 8)
xil.c2.average =
matrix(c(data.c2.average$data.X1,data.c2.average$data.X2,

```

```

data.c2.average$data.X3,data.c2.average$data.X4,
data.c2.average$data.X5,data.c2.average$data.X6,
data.c2.average$data.X7,data.c2.average$data.X8), ncol = 8)
xil.c3.average =
matrix(c (data.c3.average$data.X1,data.c3.average$data.X2,
data.c3.average$data.X3,data.c3.average$data.X4,
data.c3.average$data.X5,data.c3.average$data.X6,
data.c3.average$data.X7,data.c3.average$data.X8), ncol = 8)
xil.c4.average =
matrix(c (data.c4.average$data.X1,data.c4.average$data.X2,
data.c4.average$data.X3,data.c4.average$data.X4,
data.c4.average$data.X5,data.c4.average$data.X6,
data.c4.average$data.X7,data.c4.average$data.X8), ncol = 8)
xil.c5.average =
matrix(c (data.c5.average$data.X1,data.c5.average$data.X2,
data.c5.average$data.X3,data.c5.average$data.X4,
data.c5.average$data.X5,data.c5.average$data.X6,
data.c5.average$data.X7,data.c5.average$data.X8), ncol = 8)
xil.c6.average =
matrix(c (data.c6.average$data.X1,data.c6.average$data.X2,
data.c6.average$data.X3,data.c6.average$data.X4,
data.c6.average$data.X5,data.c6.average$data.X6,
data.c6.average$data.X7,data.c6.average$data.X8), ncol = 8)
xil.c7.average =
matrix(c (data.c7.average$data.X1,data.c7.average$data.X2,
data.c7.average$data.X3,data.c7.average$data.X4,
data.c7.average$data.X5,data.c7.average$data.X6,
data.c7.average$data.X7,data.c7.average$data.X8), ncol = 8)
xil.c8.average =
matrix(c (data.c8.average$data.X1,data.c8.average$data.X2,
data.c8.average$data.X3,data.c8.average$data.X4,
data.c8.average$data.X5,data.c8.average$data.X6,
data.c8.average$data.X7,data.c8.average$data.X8), ncol = 8)
x1.average = sum ((xil.c1.average-xlc1bar.average)^2)
x2.average = sum ((xil.c2.average-xlc2bar.average)^2)
x3.average = sum ((xil.c3.average-xlc3bar.average)^2)
x4.average = sum ((xil.c4.average-xlc4bar.average)^2)
x5.average = sum ((xil.c5.average-xlc5bar.average)^2)
x6.average = sum ((xil.c6.average-xlc6bar.average)^2)
x7.average = sum ((xil.c7.average-xlc7bar.average)^2)
x8.average = sum ((xil.c8.average-xlc8bar.average)^2)
SSW.average.c1 = sum (x1.average)
SSW.average.c2 = sum (x1.average,x2.average)
SSW.average.c3 = sum (x1.average,x2.average,x3.average)
SSW.average.c4 = sum
(x1.average,x2.average,x3.average,x4.average)
SSW.average.c5 = sum
(x1.average,x2.average,x3.average,x4.average,x5.average)
SSW.average.c6 = sum
(x1.average,x2.average,x3.average,x4.average,x5.average,x6.average)
SSW.average.c7 = sum
(x1.average,x2.average,x3.average,x4.average,x5.average,x6.average,x7.average)
SSW.average.c8 = sum
(x1.average,x2.average,x3.average,x4.average,x5.average,x6.average,x7.average,x8.average)

```

```

SSB.average.c1      = SST - SSW.average.c1
SSB.average.c2      = SST - SSW.average.c2
SSB.average.c3      = SST - SSW.average.c3
SSB.average.c4      = SST - SSW.average.c4
SSB.average.c5      = SST - SSW.average.c5
SSB.average.c6      = SST - SSW.average.c6
SSB.average.c7      = SST - SSW.average.c7
SSB.average.c8      = SST - SSW.average.c8
R.square.average.c1 = SSB.average.c1/SST
R.square.average.c2 = SSB.average.c2/SST
R.square.average.c3 = SSB.average.c3/SST
R.square.average.c4 = SSB.average.c4/SST
R.square.average.c5 = SSB.average.c5/SST
R.square.average.c6 = SSB.average.c6/SST
R.square.average.c7 = SSB.average.c7/SST
R.square.average.c8 = SSB.average.c8/SST
R.square.average.c8 = SSB.average.c8/SST
Hasil.average = c(SST, SSW.average.c8, R.square.average.c8)
Hasil = rbind(Hasil.single, Hasil.complete, Hasil.average)
print("Hasil Pengelompokan Hirarki")
print(hasil.kelompok.numerik)
print("-----")
print("  Untuk 8 kelompok yang terbentuk  ")
print("Metode      SST      SSW      R-square")
print(Hasil)
}

```

Lampiran 15. Syntax Rangkuman Nilai *R-Square* Analisis Kelompok Hirarki Agglomerative

```
Rangkuman.RSquare = function(A2, A3, A4, A5, A6, A7, A8){
  SST.2 = c(A2[,1])
  SST.3 = c(A3[,1])
  SST.4 = c(A4[,1])
  SST.5 = c(A5[,1])
  SST.6 = c(A6[,1])
  SST.7 = c(A7[,1])
  SST.8 = c(A8[,1])
  SST = rbind(SST.2,SST.3,SST.4,SST.5,SST.6,SST.7,SST.8)
  SSW.2 = c(A2[,2])
  SSW.3 = c(A3[,2])
  SSW.4 = c(A4[,2])
  SSW.5 = c(A5[,2])
  SSW.6 = c(A6[,2])
  SSW.7 = c(A7[,2])
  SSW.8 = c(A8[,2])
  SSW = rbind(SSW.2,SSW.3,SSW.4,SSW.5,SSW.6,SSW.7,SSW.8)
  SSB.2 = SST.2 - SSW.2
  SSB.3 = SST.3 - SSW.3
  SSB.4 = SST.4 - SSW.4
  SSB.5 = SST.5 - SSW.5
  SSB.6 = SST.6 - SSW.6
  SSB.7 = SST.7 - SSW.7
  SSB.8 = SST.8 - SSW.8
  SSB = rbind(SSB.2,SSB.3,SSB.4,SSB.5,SSB.6,SSB.7,SSB.8)
  Rangkuman.R.square.2 = c(A2[,3])
  Rangkuman.R.square.3 = c(A3[,3])
  Rangkuman.R.square.4 = c(A4[,3])
  Rangkuman.R.square.5 = c(A5[,3])
  Rangkuman.R.square.6 = c(A6[,3])
  Rangkuman.R.square.7 = c(A7[,3])
  Rangkuman.R.square.8 = c(A8[,3])
  Rangkuman.R.square =
  rbind(Rangkuman.R.square.1,Rangkuman.R.square.2,Rangkuman.R.square.3,
  Rangkuman.R.square.4,Rangkuman.R.square.5,Rangkuman.R.square.6,
  Rangkuman.R.square.7,Rangkuman.R.square.8)
  Rangkuman.R.square.Single = Rangkuman.R.square[,1]
  Rangkuman.R.square.Complete = Rangkuman.R.square[,2]
  Rangkuman.R.square.Average = Rangkuman.R.square[,3]
  plot(Rangkuman.R.square.Single, main = " Plot R-Square Metode
  Single Linkage",xlab = "Jumlah Kelompok")
  lines(Rangkuman.R.square.Single, col = "Red")
  plot(Rangkuman.R.square.Complete, main = " Plot R-Square Metode
  Complete Linkage",xlab = "Jumlah Kelompok")
  lines(Rangkuman.R.square.Complete, col = "Blue")
  plot(Rangkuman.R.square.Average, main = " Plot R-Square Metode
  Average Linkage",xlab = "Jumlah Kelompok")
  lines(Rangkuman.R.square.Average, col = "Green")
}
```

Lampiran 16. *Syntax* Rangkuman Nilai F Analisis Kelompok Hirarki Agglomerative

```
Rangkuman.F = function(A2, A3, A4, A5, A6, A7, A8){
  SST.2 = c(A2[,1])
  SST.3 = c(A3[,1])
  SST.4 = c(A4[,1])
  SST.5 = c(A5[,1])
  SST.6 = c(A6[,1])
  SST.7 = c(A7[,1])
  SST.8 = c(A8[,1])
  SST = rbind(SST.2, SST.3, SST.4, SST.5, SST.6, SST.7, SST.8)
  SSW.2 = c(A2[,2])
  SSW.3 = c(A3[,2])
  SSW.4 = c(A4[,2])
  SSW.5 = c(A5[,2])
  SSW.6 = c(A6[,2])
  SSW.7 = c(A7[,2])
  SSW.8 = c(A8[,2])
  SSW = rbind(SSW.2, SSW.3, SSW.4, SSW.5, SSW.6, SSW.7, SSW.8)
  SSB.2 = SST.2 - SSW.2
  SSB.3 = SST.3 - SSW.3
  SSB.4 = SST.4 - SSW.4
  SSB.5 = SST.5 - SSW.5
  SSB.6 = SST.6 - SSW.6
  SSB.7 = SST.7 - SSW.7
  SSB.8 = SST.8 - SSW.8
  SSB = rbind(SSB.2, SSB.3, SSB.4, SSB.5, SSB.6, SSB.7, SSB.8)
  F.2 = (SSB.2/1) / (SSW.2/(24-1))
  F.3 = (SSB.3/2) / (SSW.3/(24-2))
  F.4 = (SSB.4/3) / (SSW.4/(24-3))
  F.5 = (SSB.5/4) / (SSW.5/(24-4))
  F.6 = (SSB.6/5) / (SSW.6/(24-5))
  F.7 = (SSB.7/6) / (SSW.7/(24-6))
  F.8 = (SSB.8/7) / (SSW.8/(24-7))
  i = c(2,3,4,5,6,7,8)
  F = rbind(F.2, F.3, F.4, F.5, F.6, F.7, F.8)
  F.Single = cbind(F[,1])
  F.Complete = cbind(F[,2])
  F.Average = cbind(F[,3])
  F.Metode.Single = matrix(1, nrow = 7, ncol = 1)
  F.Metode.Complete = matrix(2, nrow = 7, ncol = 1)
  F.Metode.Average = matrix(3, nrow = 7, ncol = 1)
  Metode = c(F.Metode.Single, F.Metode.Complete,
  F.Metode.Average)
  Jumlah.Kelompok = c(i,i,i)
  Nilai.F = c(F.Single, F.Complete, F.Average)
  Rangkuman.F = data.frame(Metode, Jumlah.Kelompok,
  Nilai.F)
  Metode = as.numeric(Metode)
  nmetode = max(Metode)
  xrange = range(Jumlah.Kelompok)
  yrange = range(Nilai.F)
  windows()
  plot(xrange, yrange, type="n", xlab="Jumlah
  Kelompok", ylab="Nilai F" )
}
```

```

colors      = rainbow(nmetode)
linetype    = c(1:nmetode)
plotchar    = seq(18,18+nmetode,1)
for (i in 1:nmetode) {
  metode     = subset(Rangkuman.F, Metode==i)
  lines(metode$Jumlah.Kelompok, metode$Nilai.F, type="b",
        lwd=1.5,
        lty=linetype[i], col=colors[i], pch=plotchar[i])
}
title("Plot nilai F")
legend(xrange[1], yrange[2], c("Single", "Complete", "Average"),
      cex=0.8, col=colors,
      pch=plotchar, lty=linetype, title="Metode")
}

```

Lampiran 17. *Syntax* Nilai Ratio Analisis Kelompok Hirarki Agglomerative

```
Ratio.Sw.Sb.k4 = function(data) {
  k
    = 4
  datanumerik = data.frame(data$X1, data$X2, data$X3, data$X4,
    data$X5, data$X6, data$X7, data$X8)
  d
    = dist(datanumerik, method = "euclidean")
  # Analisis Kluster Hirarki
  fit.sin
    = hclust(d, method = "single")
  fit.com
    = hclust(d, method = "complete")
  fit.ave
    = hclust(d, method = "average")
  # Memotong Dendrogram untuk k Kluster
  single
    = cutree(fit.sin, k=k)
  complete
    = cutree(fit.com, k=k)
  average
    = cutree(fit.ave, k=k)
  hasil.kelompok.numerik =
  data.frame>Nama,single,complete,average)
  datanumerik2 = cbind(data$X1, data$X2, data$X3, data$X4,
    data$X5, data$X6, data$X7, data$X8)
  data1
    = mean(datanumerik2[1,])
  data2
    = mean(datanumerik2[2,])
  data3
    = mean(datanumerik2[3,])
  data4
    = mean(datanumerik2[4,])
  data5
    = mean(datanumerik2[5,])
  data6
    = mean(datanumerik2[6,])
  data7
    = mean(datanumerik2[7,])
  data8
    = mean(datanumerik2[8,])
  data9
    = mean(datanumerik2[9,])
  data10
    = mean(datanumerik2[10,])
  data11
    = mean(datanumerik2[11,])
  data12
    = mean(datanumerik2[12,])
  data13
    = mean(datanumerik2[13,])
  data14
    = mean(datanumerik2[14,])
  data15
    = mean(datanumerik2[15,])
  data16
    = mean(datanumerik2[16,])
  data17
    = mean(datanumerik2[17,])
  data18
    = mean(datanumerik2[18,])
  data19
    = mean(datanumerik2[19,])
  data20
    = mean(datanumerik2[20,])
  data21
    = mean(datanumerik2[21,])
  data22
    = mean(datanumerik2[22,])
  data23
    = mean(datanumerik2[23,])
  data24
    = mean(datanumerik2[24,])
  data25
    = mean(datanumerik2[25,])
  datarata
    = rbind(data1,data2,data3,data4,data5,data6,data7,
    data8,data9,data10,data11,data12,data13,data14,data15,data16,
    data17,data18,data19,data20,data21,data22,data23,data24,data25)
  #Analisis single linkage
  kelompok.single
    = hasil.kelompok.numerik$single
  data.single
    = data.frame(single, datarata)
  data.single.sort
    = data.single[order(data.single$single),]
  #Menghitung Sw
  mean.c1.single
    = mean(data.c1.single$datarata)
  mean.c2.single
    = mean(data.c2.single$datarata)
  mean.c3.single
    = mean(data.c3.single$datarata)
  mean.c4.single
    = mean(data.c4.single$datarata)
  s1.single
    = sqrt((sum((data.c1.single$datarata-
    mean.c1.single)^2))/(a))
}
```

```

s2.single = sqrt((sum((data.c2.single$datarata-
mean.c2.single)^2))/(b))
s3.single = sqrt((sum((data.c3.single$datarata-
mean.c3.single)^2))/(c))
s4.single = sqrt((sum((data.c4.single$datarata-
mean.c4.single)^2))/(d))
jumlah.s.single = sum(s1.single,s2.single,s3.single,s4.single)
Sw.single = jumlah.s.single/k
#Menghitung Sb
y1.single = mean.c1.single-mean(datarata)
y2.single = mean.c2.single-mean(datarata)
y3.single = mean.c3.single-mean(datarata)
y4.single = mean.c4.single-mean(datarata)
jumlah.single = sum(y1.single,y2.single,y3.single,y4.single)
Sb.single = sqrt(jumlah.single/(k-1))
#Ratio Perbandingan Sw dan Sb
Ratio.single = Sw.single/Sb.single
Hasil.single = c(Sw.single, Sb.single, Ratio.single)
#Analisis complete linkage
kelompok.complete = hasil.kelompok.numerik$complete
data.complete = data.frame(complete, datarata)
data.complete.sort =
data.complete[order(data.complete$complete),]
#Menghitung Sw
mean.c1.complete = mean(data.c1.complete$datarata)
mean.c2.complete = mean(data.c2.complete$datarata)
mean.c3.complete = mean(data.c3.complete$datarata)
mean.c4.complete = mean(data.c4.complete$datarata)
s1.complete = sqrt((sum((data.c1.complete$datarata-
mean.c1.complete)^2))/(a))
s2.complete = sqrt((sum((data.c2.complete$datarata-
mean.c2.complete)^2))/(b))
s3.complete = sqrt((sum((data.c3.complete$datarata-
mean.c3.complete)^2))/(c))
s4.complete = sqrt((sum((data.c4.complete$datarata-
mean.c4.complete)^2))/(d))
jumlah.s.complete =
sum(s1.complete,s2.complete,s3.complete,s4.complete)
Sw.complete = jumlah.s.complete/k
#Menghitung Sb
y1.complete = mean.c1.complete-mean(datarata)
y2.complete = mean.c2.complete-mean(datarata)
y3.complete = mean.c3.complete-mean(datarata)
y4.complete = mean.c4.complete-mean(datarata)
jumlah.complete =
sum(y1.complete,y2.complete,y3.complete,y4.complete)
Sb.complete = sqrt(jumlah.complete/(k-1))
#Ratio Perbandingan Sw dan Sb
Ratio.complete = Sw.complete/Sb.complete
Hasil.complete = c(Sw.complete, Sb.complete, Ratio.complete)
#Analisis average linkage
kelompok.average = hasil.kelompok.numerik$average
data.average = data.frame(average, datarata)
data.average.sort = data.average[order(data.average$average),]
#Menghitung Sw
mean.c1.average = mean(data.c1.average$datarata)
mean.c2.average = mean(data.c2.average$datarata)

```



```

mean.c3.average = mean(data.c3.average$datarata)
mean.c4.average = mean(data.c4.average$datarata)
s1.average = sqrt((sum((data.c1.average$datarata-
mean.c1.average)^2))/(a))
s2.average = sqrt((sum((data.c2.average$datarata-
mean.c2.average)^2))/(b))
s3.average = sqrt((sum((data.c3.average$datarata-
mean.c3.average)^2))/(c))
s4.average = sqrt((sum((data.c4.average$datarata-
mean.c4.average)^2))/(d))
jumlah.s.average =
sum(s1.average,s2.average,s3.average,s4.average)
Sw.average = jumlah.s.average/k
#Menghitung Sb
y1.average = mean.c1.average-mean(datarata)
y2.average = mean.c2.average-mean(datarata)
y3.average = mean.c3.average-mean(datarata)
y4.average = mean.c4.average-mean(datarata)
jumlah.average =
sum(y1.average,y2.average,y3.average,y4.average)
Sb.average = sqrt(jumlah.average/(k-1))
#Ratio Perbandingan Sw dan Sb
Ratio.average = Sw.average/Sb.average
Hasil.average = c(Sw.average, Sb.average, Ratio.average)
Hasil = rbind(Hasil.single, Hasil.complete, Hasil.average)
print("Nilai Ratio antara Sw dan Sb")
print("-----")
print("  Untuk 4 kelompok yang terbentuk  ")
print("Metode      Sw      Sb      Ratio")
print(Hasil)
}

```

Lampiran 18. Syntax Analisis Metode ROCK untuk Data Kategorik

```
Analisis.ROCK = function(data){
  datakategorik1 = c(data$X9, data$X10, data$X11, data$X12,
  data$X13, data$X14, data$X15)
  datakategorik2 = matrix (datakategorik1,25,7)
  x      = as.dummy(datakategorik2)
  rc.05 = rockCluster(x, n=5, theta=0.05, debug=FALSE)
  rc.07 = rockCluster(x, n=5, theta=0.07, debug=FALSE)
  rc.10 = rockCluster(x, n=5, theta=0.10, debug=FALSE)
  rc.12 = rockCluster(x, n=5, theta=0.12, debug=FALSE)
  rc.15 = rockCluster(x, n=5, theta=0.15, debug=FALSE)
  rc.17 = rockCluster(x, n=5, theta=0.17, debug=FALSE)
  rc.20 = rockCluster(x, n=5, theta=0.20, debug=FALSE)
  rc.22 = rockCluster(x, n=5, theta=0.22, debug=FALSE)
  rc.25 = rockCluster(x, n=5, theta=0.25, debug=FALSE)
  rc.27 = rockCluster(x, n=5, theta=0.27, debug=FALSE)
  rc.30 = rockCluster(x, n=5, theta=0.30, debug=FALSE)
  rf.05 = fitted(rc.05)
  rf.07 = fitted(rc.07)
  rf.10 = fitted(rc.10)
  rf.12 = fitted(rc.12)
  rf.15 = fitted(rc.15)
  rf.17 = fitted(rc.17)
  rf.20 = fitted(rc.20)
  rf.22 = fitted(rc.22)
  rf.25 = fitted(rc.25)
  rf.27 = fitted(rc.27)
  rf.30 = fitted(rc.30)
  theta.05 = rf.05$c1
  theta.07 = rf.07$c1
  theta.10 = rf.10$c1
  theta.12 = rf.12$c1
  theta.15 = rf.15$c1
  theta.17 = rf.17$c1
  theta.20 = rf.20$c1
  theta.22 = rf.22$c1
  theta.25 = rf.25$c1
  theta.27 = rf.27$c1
  theta.30 = rf.30$c1
  Kelompok = data.frame( theta.05, theta.07,theta.10,theta.12,
  theta.15, theta.17, theta.20,theta.22, theta.25,
  theta.27,theta.30)
  Hasil = data.frame>Nama,Kelompok)
  jumlah.05 = count(Hasil,"theta.05")
  jumlah.07 = count(Hasil,"theta.07")
  jumlah.10 = count(Hasil,"theta.10")
  jumlah.12 = count(Hasil,"theta.12")
  jumlah.15 = count(Hasil,"theta.15")
  jumlah.17 = count(Hasil,"theta.17")
  jumlah.20 = count(Hasil,"theta.20")
  jumlah.22 = count(Hasil,"theta.22")
  jumlah.25 = count(Hasil,"theta.25")
  jumlah.27 = count(Hasil,"theta.27")
  jumlah.30 = count(Hasil,"theta.30")
  X9  = factor (data$X9)
  X10 = factor (data$X10)
  X11 = factor (data$X11)
```

```

X12 = factor (data$X12)
X13 = factor (data$X13)
X14 = factor (data$X14)
X15 = factor (data$X15)
p.07 = theta.07~X9+X10+X11+X12+X13+X14+X15
p.10 = theta.10~X9+X10+X11+X12+X13+X14+X15
p.12 = theta.12~X9+X10+X11+X12+X13+X14+X15
p.15 = theta.15~X9+X10+X11+X12+X13+X14+X15
p.17 = theta.17~X9+X10+X11+X12+X13+X14+X15
p.20 = theta.20~X9+X10+X11+X12+X13+X14+X15
p.22 = theta.22~X9+X10+X11+X12+X13+X14+X15
p.25 = theta.25~X9+X10+X11+X12+X13+X14+X15
p.27 = theta.27~X9+X10+X11+X12+X13+X14+X15
mylogit.07 = aov(p.07)
mylogit.10 = aov(p.10)
mylogit.12 = aov(p.12)
mylogit.15 = aov(p.15)
mylogit.17 = aov(p.17)
mylogit.20 = aov(p.20)
mylogit.22 = aov(p.22)
mylogit.25 = aov(p.25)
mylogit.27 = aov(p.27)
SSW.07 = sum((mylogit.07$residuals)^2)
SSW.10 = sum((mylogit.10$residuals)^2)
SSW.12 = sum((mylogit.12$residuals)^2)
SSW.15 = sum((mylogit.15$residuals)^2)
SSW.17 = sum((mylogit.17$residuals)^2)
SSW.20 = sum((mylogit.20$residuals)^2)
SSW.22 = sum((mylogit.22$residuals)^2)
SSW.25 = sum((mylogit.25$residuals)^2)
SSW.27 = sum((mylogit.27$residuals)^2)
summary.07 = c(summary(mylogit.07))
summary.10 = c(summary(mylogit.10))
summary.12 = c(summary(mylogit.12))
summary.15 = c(summary(mylogit.15))
summary.17 = c(summary(mylogit.17))
summary.20 = c(summary(mylogit.20))
summary.22 = c(summary(mylogit.22))
summary.25 = c(summary(mylogit.25))
summary.27 = c(summary(mylogit.27))
SSB.07 = sum(summary.07[1:7,3])
SSB.10 = sum(summary.10[1:7,3])
SSB.12 = sum(summary.12[1:7,3])
SSB.15 = sum(summary.15[1:7,3])
SSB.17 = sum(summary.17[1:7,3])
SSB.20 = sum(summary.20[1:7,3])
SSB.22 = sum(summary.22[1:7,3])
SSB.25 = sum(summary.25[1:7,3])
SSB.27 = sum(summary.27[1:7,3])
SW.07 = sqrt(SSW.07/(25-2))
SW.10 = sqrt(SSW.10/(25-2))
SW.12 = sqrt(SSW.12/(25-2))
SW.15 = sqrt(SSW.15/(25-3))
SW.17 = sqrt(SSW.17/(25-3))
SW.20 = sqrt(SSW.20/(25-3))
SW.22 = sqrt(SSW.22/(25-2))
SW.25 = sqrt(SSW.25/(25-1))

```

```

SW.27 = sqrt(SSW.27/(25-2))
SB.07 = sqrt(SSB.07/(2-1))
SB.10 = sqrt(SSB.10/(2-1))
SB.12 = sqrt(SSB.12/(2-1))
SB.15 = sqrt(SSB.15/(3-1))
SB.17 = sqrt(SSB.17/(3-1))
SB.20 = sqrt(SSB.20/(3-1))
SB.22 = sqrt(SSB.22/(2-1))
SB.25 = sqrt(SSB.25/(1-1))
SB.27 = sqrt(SSB.27/(2-1))
Ratio.07 = SW.07/SB.07
Ratio.10 = SW.10/SB.10
Ratio.12 = SW.12/SB.12
Ratio.15 = SW.15/SB.15
Ratio.17 = SW.17/SB.17
Ratio.20 = SW.20/SB.20
Ratio.22 = SW.22/SB.22
Ratio.25 = SW.25/SB.25
Ratio.27 = SW.27/SB.27
Ratio.kategorik = rbind(Ratio.07, Ratio.10, Ratio.12, Ratio.15,
Ratio.17, Ratio.20, Ratio.22, Ratio.25, Ratio.27)
print("Hasil Pengelompokan ROCK")
print(Hasil)
print("-----")
print(jumlah.05)
print(jumlah.07)
print(jumlah.10)
print(jumlah.12)
print(jumlah.15)
print(jumlah.17)
print(jumlah.20)
print(jumlah.22)
print(jumlah.25)
print(jumlah.27)
print(jumlah.30)
print("-----")
print("Nilai Ratio yanga Terbentuk")
print(Ratio.kategorik)
u = c(0.07,0.10,0.12,0.15,0.17,0.20,0.22,0.25,0.27)
plot(u, z, main = " Plot Nilai Ratio Metode ROCK",
xlab = "Nilai Theta", ylab = "Nilai Ratio")
lines(u,z,col = "Red")
}

```

Lampiran 19. *Syntax* Analisis Pengelompokan Ensemble ROCK

```
Ensemble.ROCK = function(data, k.numerik, k.kategorik,
theta.kategorik){
  datanumerik      = data.frame(data$X1, data$X2, data$X3,
data$X4, data$X5, data$X6, data$X7, data$X8)
  datakategorik    = data.frame(data$X9, data$X10, data$X11,
data$X12, data$X13, data$X14, data$X15)
  # Pengelompokan Numerik
  d      = dist(datanumerik, method = "euclidean")
  fit.sin = hclust(d, method = "single")
  k = k.numerik
  single = cutree(fit.sin, k=k)
  # Pengelompokan Kategorik
  datakategorik1 = c(data$X9, data$X10, data$X11, data$X12,
data$X13, data$X14, data$X15)
  datakategorik2 = matrix (datakategorik1,25,7)
  d      = as.dummy(datakategorik2)
  n      = k.kategorik
  t      = theta.kategorik
  rc     = rockCluster(d,n=n,theta=t,debug=TRUE)
  rf.hasil = fitted(rc)
  x      = rf.hasil$c1
  # Pengelompokan Ensemble ROCK
  datakategorik = data.frame(single,x)
  datakategorik1 = c(datakategorik$single, datakategorik$x)
  datakategorik2 = matrix (datakategorik1,25,2)
  Hasil.Ensemble.ROCK = Analisis.ROCK2 (datakategorik2)
  # Menghitung Ratio
  d = as.dummy(datakategorik2)
  rc.07 = rockCluster(d,n=5,theta=0.07,funArgs=NULL, debug=FALSE)
  rc.10 = rockCluster(d,n=5,theta=0.10,funArgs=NULL, debug=FALSE)
  rc.12 = rockCluster(d,n=5,theta=0.12,funArgs=NULL, debug=FALSE)
  rc.15 = rockCluster(d,n=5,theta=0.15,funArgs=NULL, debug=FALSE)
  rc.17 = rockCluster(d,n=5,theta=0.17,funArgs=NULL, debug=FALSE)
  rc.20 = rockCluster(d,n=5,theta=0.20,funArgs=NULL, debug=FALSE)
  rc.22 = rockCluster(d,n=5,theta=0.22,funArgs=NULL, debug=FALSE)
  rc.25 = rockCluster(d,n=5,theta=0.25,funArgs=NULL, debug=FALSE)
  rc.27 = rockCluster(d,n=5,theta=0.27,funArgs=NULL, debug=FALSE)
  rc.30 = rockCluster(d,n=5,theta=0.30,funArgs=NULL, debug=FALSE)
  rf.07 = fitted(rc.07)
  rf.10 = fitted(rc.10)
  rf.12 = fitted(rc.12)
  rf.15 = fitted(rc.15)
  rf.17 = fitted(rc.17)
  rf.20 = fitted(rc.20)
  rf.22 = fitted(rc.22)
  rf.25 = fitted(rc.25)
  rf.27 = fitted(rc.27)
  rf.30 = fitted(rc.30)
  theta.07 = rf.07$c1
  theta.10 = rf.10$c1
  theta.12 = rf.12$c1
  theta.15 = rf.15$c1
  theta.17 = rf.17$c1
  theta.20 = rf.20$c1
  theta.22 = rf.22$c1
  theta.25 = rf.25$c1
```

```

theta.27 = rf.27$c1
theta.30 = rf.30$c1
Kelompok = data.frame(theta.07,theta.10,theta.12, theta.15,
theta.17, theta.20,theta.22, theta.25, theta.27, theta.30)
Hasil = data.frame>Nama,datakategorik,Kelompok)
Data.Ensembl = Hasil
p.07 = theta.07~X1+X2
p.10 = theta.10~X1+X2
p.12 = theta.12~X1+X2
p.15 = theta.15~X1+X2
p.17 = theta.17~X1+X2
p.20 = theta.20~X1+X2
p.22 = theta.22~X1+X2
p.25 = theta.25~X1+X2
p.27 = theta.27~X1+X2
p.30 = theta.30~X1+X2
mylogit.07 = aov(p.07)
mylogit.10 = aov(p.10)
mylogit.12 = aov(p.12)
mylogit.15 = aov(p.15)
mylogit.17 = aov(p.17)
mylogit.20 = aov(p.20)
mylogit.22 = aov(p.22)
mylogit.25 = aov(p.25)
mylogit.27 = aov(p.27)
mylogit.30 = aov(p.30)
SSW.07 = sum((mylogit.07$residuals)^2)
SSW.10 = sum((mylogit.10$residuals)^2)
SSW.12 = sum((mylogit.12$residuals)^2)
SSW.15 = sum((mylogit.15$residuals)^2)
SSW.17 = sum((mylogit.17$residuals)^2)
SSW.20 = sum((mylogit.20$residuals)^2)
SSW.22 = sum((mylogit.22$residuals)^2)
SSW.25 = sum((mylogit.25$residuals)^2)
SSW.27 = sum((mylogit.27$residuals)^2)
SSW.30 = sum((mylogit.30$residuals)^2)
summary1.07 = c(summary(mylogit.07))
summary1.10 = c(summary(mylogit.10))
summary1.12 = c(summary(mylogit.12))
summary1.15 = c(summary(mylogit.15))
summary1.17 = c(summary(mylogit.17))
summary1.20 = c(summary(mylogit.20))
summary1.22 = c(summary(mylogit.22))
summary1.25 = c(summary(mylogit.25))
summary1.27 = c(summary(mylogit.27))
summary1.30 = c(summary(mylogit.30))
SSB.07 = sum(summary1.07[1:2,3])
SSB.10 = sum(summary1.10[1:2,3])
SSB.12 = sum(summary1.12[1:2,3])
SSB.15 = sum(summary1.15[1:2,3])
SSB.17 = sum(summary1.17[1:2,3])
SSB.20 = sum(summary1.20[1:2,3])
SSB.22 = sum(summary1.22[1:2,3])
SSB.25 = sum(summary1.25[1:2,3])
SSB.27 = sum(summary1.27[1:2,3])
SSB.30 = sum(summary1.30[1:2,3])
SW.07 = sqrt(SSW.07/(25-4))

```

```

SW.10 = sqrt(SSW.10/(25-4))
SW.12 = sqrt(SSW.12/(25-4))
SW.15 = sqrt(SSW.15/(25-4))
SW.17 = sqrt(SSW.17/(25-4))
SW.20 = sqrt(SSW.20/(25-4))
SW.22 = sqrt(SSW.22/(25-4))
SW.25 = sqrt(SSW.25/(25-3))
SW.27 = sqrt(SSW.27/(25-3))
SW.30 = sqrt(SSW.30/(25-3))
SB.07 = sqrt(SSB.07/(4-1))
SB.10 = sqrt(SSB.10/(4-1))
SB.12 = sqrt(SSB.12/(4-1))
SB.15 = sqrt(SSB.15/(4-1))
SB.17 = sqrt(SSB.17/(4-1))
SB.20 = sqrt(SSB.20/(4-1))
SB.22 = sqrt(SSB.22/(4-1))
SB.25 = sqrt(SSB.25/(3-1))
SB.27 = sqrt(SSB.27/(3-1))
SB.30 = sqrt(SSB.30/(3-1))
Ratio.07 = (SW.07/SB.07)
Ratio.10 = (SW.10/SB.10)
Ratio.12 = (SW.12/SB.12)
Ratio.15 = (SW.15/SB.15)
Ratio.17 = (SW.17/SB.17)
Ratio.20 = (SW.20/SB.20)
Ratio.22 = (SW.22/SB.22)
Ratio.25 = (SW.25/SB.25)
Ratio.27 = (SW.27/SB.27)
Ratio.30 = (SW.30/SB.30)
Ratio.Ensembl.ROCK = rbind(Ratio.07, Ratio.10, Ratio.12,
Ratio.15, Ratio.17,Ratio.20, Ratio.22, Ratio.25, Ratio.27,
Ratio.30)
plot(u, z, main = " Plot Nilai Ratio Ensembl ROCK",
xlab = "Nilai Theta", ylab = "Nilai Ratio")
lines(u,z,col = "Blue")
print("-----")
print("Nilai Ratio untuk Setiap Nilai Theta")
print(Ratio.Ensembl.ROCK)
}

```

Lampiran 20. Syntax Analisis Pengelompokan Ensemble SWFM

```
Ensemble.SWFM = function(data, k.numerik, k.kategorik,
theta.kategorik){
  datanumerik      = data.frame(data$X1, data$X2, data$X3,
data$X4, data$X5, data$X6, data$X7, data$X8)
  datakategorik    = data.frame(data$X9, data$X10, data$X11,
data$X12, data$X13, data$X14, data$X15)
  # Pengelompokan Numerik
  d      = dist(datanumerik, method = "euclidean")
  fit.sin = hclust(d, method = "single")
  k = k.numerik
  single = cutree(fit.sin, k=k)
  # Pengelompokan Kategorik
  datakategorik1 = c(data$X9, data$X10, data$X11, data$X12,
data$X13, data$X14, data$X15)
  datakategorik2 = matrix (datakategorik1,25,7)
  d      = as.dummy(datakategorik2)
  n      = k.kategorik
  t      = theta.kategorik
  rc     = rockCluster(d,n=n,theta=t,debug=TRUE)
  rf.hasil = fitted(rc)
  u      = rf.hasil$cl
  # Pengelompokan Ensemble SWFM
  #Menghitung jarak (Similarity Weight)
  sij = function(x,y){
  sij = length(intersect(x,y))/length(union(x,y))
  p = length(unique(x))
  q = length(unique(y))
  m = max (p,q)
  sm = sum (sij/m)
  return(sm)
  }
  data1 = c(single[1],u[1])
  data2 = c(single[2],u[2])
  data3 = c(single[3],u[3])
  data4 = c(single[4],u[4])
  data5 = c(single[5],u[5])
  data6 = c(single[6],u[6])
  data7 = c(single[7],u[7])
  data8 = c(single[8],u[8])
  data9 = c(single[9],u[9])
  data10 = c(single[10],u[10])
  data11 = c(single[11],u[11])
  data12 = c(single[12],u[12])
  data13 = c(single[13],u[13])
  data14 = c(single[14],u[14])
  data15 = c(single[15],u[15])
  data16 = c(single[16],u[16])
  data17 = c(single[17],u[17])
  data18 = c(single[18],u[18])
  data19 = c(single[19],u[19])
  data20 = c(single[20],u[20])
  data21 = c(single[21],u[21])
  data22 = c(single[22],u[22])
  data23 = c(single[23],u[23])
  data24 = c(single[24],u[24])
  data25 = c(single[25],u[25])
}
```



```

data=
rbind(data1,data2,data3,data4,data5,data6,data7,data8,data9,
data10,data11,data12,data13,data14,data15,data16,data17,
data18,data19,data20,data21,data22,data23,data24,data25)
sim  = matrix (1, nrow = 25, ncol = 25)
rownames(sim) = c(1:25)
colnames(sim) = c(1:25)
for (i in 1:(nrow(sim)-1)){
for (j in (i+1):nrow(sim)){
x = data[i,]
y = data[j,]
sim [i,j] = sij(x,y)
sim [j,i] = sim[i,j]
}
}
wi      = 0.5
F       = wi*(1-sim)
jarak   = as.dist(F)
fit     = hclust(jarak,method = "single")
hasil.k2 = cutree(fit, k=2)
hasil.k3 = cutree(fit, k=3)
hasil.k4 = cutree(fit, k=4)
hasil.k5 = cutree(fit, k=5)
hasil.k6 = cutree(fit, k=6)
hasil.kelompok =
data.frame>Nama,hasil.k2,hasil.k3,hasil.k4,hasil.k5,hasil.k6)
SWFM = data.frame(hasil.kelompok,single,u)
write.csv(SWFM,file = "Data Hasil Ensembel SWFM.csv")
SWFM = read.csv("Data Hasil Ensembel SWFM.csv", header=TRUE,
sep=",")
p.2 = hasil.k2~X1+X2
p.3 = hasil.k3~X1+X2
p.4 = hasil.k4~X1+X2
p.5 = hasil.k5~X1+X2
p.6 = hasil.k6~X1+X2
model.2 = aov(p.2)
model.3 = aov(p.3)
model.4 = aov(p.4)
model.5 = aov(p.5)
model.6 = aov(p.6)
SSW.2 = sum((model.2$residuals)^2)
SSW.3 = sum((model.3$residuals)^2)
SSW.4 = sum((model.4$residuals)^2)
SSW.5 = sum((model.5$residuals)^2)
SSW.6 = sum((model.6$residuals)^2)
summary.2 = c(summary(model.2))
summary.3 = c(summary(model.3))
summary.4 = c(summary(model.4))
summary.5 = c(summary(model.5))
summary.6 = c(summary(model.6))
SSB.2 = sum(summary.2[1:2,3])
SSB.3 = sum(summary.3[1:2,3])
SSB.4 = sum(summary.4[1:2,3])
SSB.5 = sum(summary.5[1:2,3])
SSB.6 = sum(summary.6[1:2,3])
SW.2 = sqrt(SSW.2/(25-2))
SW.3 = sqrt(SSW.3/(25-3))

```

```

SW.4 = sqrt(SSW.4/(25-4))
SW.5 = sqrt(SSW.5/(25-5))
SW.6 = sqrt(SSW.6/(25-6))
SB.2 = sqrt(SSB.2/(2-1))
SB.3 = sqrt(SSB.3/(3-1))
SB.4 = sqrt(SSB.4/(4-1))
SB.5 = sqrt(SSB.5/(5-1))
SB.6 = sqrt(SSB.6/(6-1))
Ratio.2 = SW.2 / SB.2
Ratio.3 = SW.3 / SB.3
Ratio.4 = SW.4 / SB.4
Ratio.5 = SW.5 / SB.5
Ratio.6 = SW.6 / SB.6
Ratio.SWFM = rbind (Ratio.2, Ratio.3, Ratio.4, Ratio.5, Ratio.6)
u = c(2,3,4,5,6)
z = Ratio.SWFM
plot(u, z, main = " Plot Nilai Ratio Ensembel SWFM",
xlab = "Jumlah Kluster", ylab = "Nilai Ratio")
lines(u,z,col = "Green")
print("Hasil Pengelompokan SWFM")
print(hasil.kelompok)
print("-----")
print("Nilai Ratio untuk Setiap Nilai K")
print(Ratio.SWFM)
}

```

Lampiran 21. *Syntax* Perbandingan Hasil Pengelompokan

```
Bandung.Ratio = function (ratio1, ratio2){  
  
  x = min (ratio1)  
  y = min (ratio2)  
  
  if ( x < y){  
    hasil = "Metode Ensemble ROCK"  
  } else hasil = "Metode Ensemble SWFM"  
  
  if ( hasil == "Metode Ensemble ROCK"){  
    hasil2 = min(x)  
  } else hasil2 = min(y)  
  
  print ( "Berdasarkan nilai Ratio SW dan SB")  
  print ( "metode analisis yang terbaik adalah")  
  print ( hasil )  
  print ( "dengan nilai ratio terkecil yang dihasilkan sebesar")  
  print ( hasil2 )  
}
```

Lampiran 22. Output Software R untuk Analisis Kelompok pada Data Numerik

a. Jumlah kelompok yang dibentuk sama dengan 2

```
> Analisis.kluster.k2 (data)
[1] "Hasil Pengelompokan Hirarki"
      Nama single complete average
1    FS01      1          1         1
2    FS07      1          1         1
3    FS10      1          1         1
4    FS14      2          2         2
5    FS15      1          1         1
6    FS16      1          2         1
7    FS18      1          2         1
8    FS20      1          1         1
9    FS22      1          1         1
10   FS29      1          1         1
11   FS32      1          2         1
12   FS56      1          1         1
13   FS57      1          1         1
14   FS64      1          2         1
15   FS66      2          2         2
16   FS68      1          1         1
17   FS70      1          1         1
18   FS73      1          1         1
19   FS84      1          1         1
20   FS89      1          1         1
21   FS103     1          1         1
22   FS107     1          1         1
23 P III i     1          1         1
24   TL 3      1          1         1
25   TL 5      1          1         1
[1] "-----"
[1] " Untuk 2 kelompok yang terbentuk "
```

	SST	SSW	R-square
	[,1]	[,2]	[,3]
Hasil.single	24422.56	15633.38	0.3598795
Hasil.complete	24422.56	10830.45	0.5565390
Hasil.average	24422.56	15633.38	0.3598795

b. Jumlah kelompok yang dibentuk sama dengan 3

```
> Analisis.kluster.k3 (data)
[1] "Hasil Pengelompokan Hirarki"
      Nama single complete average
1    FS01      1          1         1
2    FS07      2          2         2
3    FS10      1          1         1
4    FS14      3          3         3
5    FS15      1          1         1
6    FS16      1          3         1
7    FS18      1          3         1
```

```

8      FS20      1      1      1
9      FS22      2      2      2
10     FS29      2      2      2
11     FS32      1      3      1
12     FS56      2      2      2
13     FS57      1      1      1
14     FS64      1      3      1
15     FS66      3      3      3
16     FS68      1      1      1
17     FS70      1      1      1
18     FS73      1      1      1
19     FS84      2      2      2
20     FS89      1      1      1
21     FS103     1      1      1
22     FS107     2      2      2
23 P III i      1      1      1
24     TL 3      2      2      2
25     TL 5      2      2      2
[1] "-----"
[1] " Untuk 3 kelompok yang terbentuk "
[1] "Metode          SST          SSW    R-square"
      [,1]      [,2]      [,3]
Hasil.single    24422.56 5096.305 0.791328
Hasil.complete  24422.56 4290.165 0.824336
Hasil.average    24422.56 5096.305 0.791328

```

c. Jumlah kelompok yang dibentuk sama dengan 4

```

> Analisis.kluster.k4 (data)
[1] "Hasil Pengelompokan Hirarki"
      Nama single complete average
1      FS01      1      1      1
2      FS07      2      2      2
3      FS10      1      1      1
4      FS14      3      3      3
5      FS15      1      1      1
6      FS16      4      4      4
7      FS18      4      4      4
8      FS20      1      1      1
9      FS22      2      2      2
10     FS29      2      2      2
11     FS32      4      4      4
12     FS56      2      2      2
13     FS57      1      1      1
14     FS64      4      4      4
15     FS66      3      3      3
16     FS68      1      1      1
17     FS70      1      1      1
18     FS73      1      1      1
19     FS84      2      2      2
20     FS89      1      1      1
21     FS103     1      1      1
22     FS107     2      2      2
23 P III i      1      1      1
24     TL 3      2      2      2
25     TL 5      2      2      2

```

```

[1] "-----"
[1] " Untuk 4 kelompok yang terbentuk "
[1] "Metode          SST          SSW    R-square"
      [,1]      [,2]      [,3]
Hasil.single  24422.56  2812.027  0.8848595
Hasil.complete 24422.56  2812.027  0.8848595
Hasil.average  24422.56  2812.027  0.8848595

```

d. Jumlah kelompok yang dibentuk sama dengan 5

```

> Analisis.kluster.k5 (data)
[1] "Hasil Pengelompokan Hirarki"
      Nama single complete average
1      FS01      1         1         1
2      FS07      2         2         2
3      FS10      1         1         1
4      FS14      3         3         3
5      FS15      1         4         4
6      FS16      4         5         5
7      FS18      4         5         5
8      FS20      1         4         4
9      FS22      2         2         2
10     FS29      2         2         2
11     FS32      5         5         5
12     FS56      2         2         2
13     FS57      1         1         4
14     FS64      4         5         5
15     FS66      3         3         3
16     FS68      1         4         4
17     FS70      1         4         4
18     FS73      1         1         1
19     FS84      2         2         2
20     FS89      1         1         4
21     FS103     1         1         4
22     FS107     2         2         2
23 P III i      1         4         4
24     TL 3      2         2         2
25     TL 5      2         2         2
[1] "-----"

[1] " Untuk 5 kelompok yang terbentuk "
[1] "Metode          SST          SSW    R-square"
      [,1]      [,2]      [,3]
Hasil.single  24422.56  2582.891  0.8942416
Hasil.complete 24422.56  2303.393  0.9056858
Hasil.average  24422.56  2197.475  0.9100228

```

e. Jumlah kelompok yang dibentuk sama dengan 6

```
> Analisis.kluster.k6 (data)
[1] "Hasil Pengelompokan Hirarki"
      Nama single complete average
1      FS01      1         1         1
2      FS07      2         2         2
3      FS10      1         1         1
4      FS14      3         3         3
5      FS15      1         4         4
6      FS16      4         5         5
7      FS18      5         5         5
8      FS20      1         4         4
9      FS22      2         2         6
10     FS29      2         2         2
11     FS32      6         5         5
12     FS56      2         2         2
13     FS57      1         6         4
14     FS64      5         5         5
15     FS66      3         3         3
16     FS68      1         4         4
17     FS70      1         4         4
18     FS73      1         1         1
19     FS84      2         2         2
20     FS89      1         6         4
21     FS103     1         6         4
22     FS107     2         2         2
23 P III i      1         4         4
24     TL 3      2         2         2
25     TL 5      2         2         2

[1] "-----"
[1] " Untuk 6 kelompok yang terbentuk "
[1] "Metode      SST      SSW  R-square"
      [,1]      [,2]      [,3]
Hasil.single 24422.56 2415.623 0.9010905
Hasil.complete 24422.56 1829.530 0.9250885
Hasil.average 24422.56 1924.443 0.9212023
```

f. Jumlah kelompok yang dibentuk sama dengan 7

```
> Analisis.kluster.k7 (data)
[1] "Hasil Pengelompokan Hirarki"
      Nama single complete average
1    FS01      1         1         1
2    FS07      2         2         2
3    FS10      1         1         1
4    FS14      3         3         3
5    FS15      1         4         4
6    FS16      4         5         5
7    FS18      5         5         6
8    FS20      1         4         4
9    FS22      6         6         7
10   FS29      2         2         2
11   FS32      7         5         5
12   FS56      2         2         2
13   FS57      1         7         4
14   FS64      5         5         6
15   FS66      3         3         3
16   FS68      1         4         4
17   FS70      1         4         4
18   FS73      1         1         1
19   FS84      2         2         2
20   FS89      1         7         4
21   FS103     1         7         4
22   FS107     2         2         2
23 P III i     1         4         4
24   TL 3      2         2         2
25   TL 5      2         2         2

[1] "-----"
[1] " Untuk 7 kelompok yang terbentuk "
```

[1] "Metode	SST	SSW	R-square
	[,1]	[,2]	[,3]
Hasil.single	24422.56	2142.591	0.9122700
Hasil.complete	24422.56	1556.498	0.9362680
Hasil.average	24422.56	1634.922	0.9330569

g. Jumlah kelompok yang dibentuk sama dengan 8

```
> Analisis.kluster.k8 (data)
[1] "Hasil Pengelompokan Hirarki"
      Nama single complete average
1    FS01      1         1         1
2    FS07      2         2         2
3    FS10      3         1         1
4    FS14      4         3         3
5    FS15      1         4         4
6    FS16      5         5         5
7    FS18      6         6         6
8    FS20      1         4         4
9    FS22      7         7         7
10   FS29      2         2         2
11   FS32      8         5         5
```


12	FS56	2	2	2
13	FS57	1	8	4
14	FS64	6	6	6
15	FS66	4	3	3
16	FS68	1	4	4
17	FS70	1	4	4
18	FS73	1	1	1
19	FS84	2	2	2
20	FS89	1	8	8
21	FS103	1	8	8
22	FS107	2	2	2
23	P III i	1	4	4
24	TL 3	2	2	2
25	TL 5	2	2	2
[1] "-----"				
[1] " Untuk 8 kelompok yang terbentuk "				
[1] "Metode SST SSW R-square"				
[,1] [,2] [,3]				
Hasil.single	24422.56	1905.811	0.9219651	
Hasil.complete	24422.56	1266.978	0.9481227	
Hasil.average	24422.56	1275.881	0.9477581	

Lampiran 23. Output Software R untuk Analisis Pengelompokan Metode ROCK pada Data Kategorik

```
> Analisis.ROCK (data)
[1] "Hasil Pengelompokan ROCK"
  X  Nama theta.05 theta.07 theta.10 theta.12 theta.15 theta.17 theta.20
1  1  FS01      5      3      4      5      4      3      2
2  2  FS07      5      4      2      5      5      3      2
3  3  FS10      5      3      4      5      2      3      2
4  4  FS14      5      4      2      5      2      3      2
5  5  FS15      5      3      5      3      5      1      1
6  6  FS16      5      3      4      5      2      3      4
7  7  FS18      5      3      4      5      4      3      4
8  8  FS20      5      4      4      5      4      1      4
9  9  FS22      5      3      4      5      2      3      4
10 10 FS29      5      3      4      5      2      2      2
11 11 FS32      5      4      4      5      4      1      2
12 12 FS56      5      4      2      5      4      3      2
13 13 FS57      5      4      2      5      4      3      4
14 14 FS64      5      3      2      5      2      1      2
15 15 FS66      5      3      4      3      5      2      2
16 16 FS68      5      4      4      5      5      3      2
17 17 FS70      5      4      2      5      4      3      2
18 18 FS73      5      3      5      5      5      2      1
19 19 FS84      5      4      4      5      2      2      4
20 20 FS89      5      4      2      5      4      1      4
21 21 FS103     5      3      2      5      4      3      2
22 22 FS107     5      3      4      5      5      3      2
23 23 PIIIIi    5      3      2      5      2      3      4
24 24 TL 3      5      3      4      5      2      1      4
25 25 TL 5      5      4      4      5      2      3      2
  theta.22 theta.25 theta.27 theta.30
1      2      1      2      1
2      2      1      2      1
3      2      1      2      1
4      2      1      2      1
5      3      1      2      1
6      2      1      2      1
7      2      1      2      1
8      3      1      1      1
9      2      1      2      1
10     2      1      2      1
11     2      1      2      1
12     3      1      2      1
13     3      1      2      1
14     2      1      2      1
15     2      1      2      1
16     2      1      2      1
17     2      1      2      1
18     3      1      1      1
19     2      1      2      1
20     3      1      1      1
21     3      1      2      1
22     2      1      2      1
23     2      1      2      1
24     2      1      2      1
25     2      1      2      1
[1] "-----"
  theta.05 freq
1      5    25
  theta.07 freq
1      3    14
```

```

2      4    11
  theta.10 freq
1      2     9
2      4    14
3      5     2
  theta.12 freq
1      3     2
2      5    23
  theta.15 freq
1      2    10
2      4     9
3      5     6
  theta.17 freq
1      1     6
2      2     4
3      3    15
  theta.20 freq
1      1     2
2      2    14
3      4     9
  theta.22 freq
1      2    18
2      3     7
  theta.25 freq
1      1    25
  theta.27 freq
1      1     3
2      2    22
  theta.30 freq
1      1    25
[1] "-----"
[1] "Nilai Ratio yang Terbentuk"
      [,1]
Ratio.07 0.1898560
Ratio.10 0.2027238
Ratio.12 0.1803070
Ratio.15 0.2504356
Ratio.17 0.1441876
Ratio.20 0.2271649
Ratio.22 0.1594033
Ratio.25 0.0000000
Ratio.27 0.1495206

```

Lampiran 24. Output Software R untuk Analisis Ensembel ROCK

```

> k.numerik= 4
> k.kategorik= 5
> theta.kategorik= 0.17
> Ensembel.ROCK(data, k.numerik, k.kategorik, theta.kategorik)

[1] "Hasil Pengelompokan ROCK"
      Nama  theta.05  theta.07  theta.10  theta.12  theta.15  theta.17
theta.20
1      FS01         2         2         2         2         2         2
2      FS07         3         3         3         3         3         3
3      FS10         2         2         2         2         2         2
4      FS14         1         1         1         1         1         1
5      FS15         4         4         4         4         4         4
6      FS16         4         4         4         4         4         4
7      FS18         4         4         4         4         4         4
8      FS20         4         4         4         4         4         4
9      FS22         3         3         3         3         3         3
10     FS29         3         3         3         3         3         3
11     FS32         4         4         4         4         4         4
12     FS56         3         3         3         3         3         3
13     FS57         2         2         2         2         2         2
14     FS64         4         4         4         4         4         4
15     FS66         3         3         3         3         3         3
16     FS68         2         2         2         2         2         2
17     FS70         2         2         2         2         2         2
18     FS73         3         2         2         2         2         2
19     FS84         4         4         4         4         4         4
20     FS89         4         4         4         4         4         4
21     FS103        2         2         2         2         2         2
22     FS107        3         3         3         3         3         3
23     PIIIIi       2         2         2         2         2         2
24     TL 3         4         4         4         4         4         4
25     TL 5         3         3         3         3         3         3
      theta.22  theta.25  theta.27  theta.30
1           2           1           1           1
2           3           5           5           5
3           2           1           1           1
4           1           1           1           1
5           4           3           3           3
6           4           3           3           3
7           4           3           3           3
8           4           5           5           5
9           3           5           5           5
10          3           5           5           5
11          4           3           3           3
12          3           5           5           5
13          2           1           1           1
14          4           3           3           5
15          3           1           1           1
16          2           1           1           1
17          2           1           1           1
18          2           1           1           1
19          4           5           3           3
20          4           3           5           5
21          2           1           1           1
22          3           5           5           5
23          2           1           1           1
24          4           3           3           5
25          3           5           5           5

[1] "-----"
      theta.05 freq

```

1	1	1
2	2	7
3	3	8
4	4	9
theta.07 freq		
1	1	1
2	2	8
3	3	7
4	4	9
theta.10 freq		
1	1	1
2	2	8
3	3	7
4	4	9
theta.12 freq		
1	1	1
2	2	8
3	3	7
4	4	9
theta.15 freq		
1	1	1
2	2	8
3	3	7
4	4	9
theta.17 freq		
1	1	1
2	2	8
3	3	7
4	4	9
theta.20 freq		
1	1	1
2	2	8
3	3	7
4	4	9
theta.22 freq		
1	1	1
2	2	8
3	3	7
4	4	9
theta.25 freq		
1	1	10
2	3	7
3	5	8
theta.27 freq		
1	1	10
2	3	7
3	5	8
theta.30 freq		
1	1	10
2	3	5
3	5	10

```
[1] "-----"
[1] "Nilai Ratio untuk Setiap Nilai Theta"
      [,1]
Ratio.07 0.1990729
Ratio.10 0.1990729
Ratio.12 0.1990729
Ratio.15 0.1990729
Ratio.17 0.1990729
Ratio.20 0.1990729
Ratio.22 0.1990729
Ratio.25 0.1731605
Ratio.27 0.1357722
Ratio.30 0.6999837
```

Lampiran 25. Output Software R untuk Analisis Ensembel SWFM

```
> k.numerik= 4
> k.kategorik= 5
> theta.kategorik= 0.17
> Ensembel.SWFM(data, k.numerik, k.kategorik, theta.kategorik)
```

[1] "Hasil Pengelompokan SWFM"

	Nama	hasil.k2	hasil.k3	hasil.k4	hasil.k5	hasil.k6
1	FS01	1	1	1	1	1
2	FS07	1	1	1	1	1
3	FS10	1	1	1	1	1
4	FS14	1	1	1	1	1
5	FS15	1	1	1	1	2
6	FS16	1	1	1	2	3
7	FS18	1	1	1	2	3
8	FS20	1	1	1	1	2
9	FS22	1	1	1	1	1
10	FS29	1	1	2	3	4
11	FS32	1	2	3	4	5
12	FS56	1	1	1	1	1
13	FS57	1	1	1	1	1
14	FS64	1	2	3	4	5
15	FS66	1	1	1	1	1
16	FS68	1	1	1	1	1
17	FS70	1	1	1	1	1
18	FS73	2	3	4	5	6
19	FS84	2	3	4	5	6
20	FS89	1	1	1	1	2
21	FS103	1	1	1	1	1
22	FS107	1	1	1	1	1
23	P III i	1	1	1	1	1
24	TL 3	2	3	4	5	6
25	TL 5	1	1	1	1	1

[1] "-----"

[1] "Nilai Ratio untuk Setiap Nilai K"

```
[,1]
Ratio.2 0.3059327
Ratio.3 0.3638819
Ratio.4 0.3596194
Ratio.5 0.3727235
Ratio.6 0.3282299
```

Lampiran 26. Nilai *R-Square* dan *F* Analisis Pengelompokan Data Numerik

a. Tabel Nilai *R-Square* berdasarkan jumlah kelompok

Jumlah Kelompok	<i>Single Linkage</i>	<i>Complete Linkage</i>	<i>Average Linkage</i>
2 Kelompok	0,3599	0,5565	0,3599
3 Kelompok	0,7913	0,8243	0,7913
4 Kelompok	0,8849	0,8849	0,8849
5 Kelompok	0,8942	0,9057	0,9100
6 Kelompok	0,9011	0,9251	0,9212
7 Kelompok	0,9123	0,9363	0,9331
8 Kelompok	0,9220	0,9481	0,9478

b. Tabel Nilai *F* berdasarkan jumlah kelompok

Jumlah Kelompok	<i>Single Linkage</i>	<i>Complete Linkage</i>	<i>Average Linkage</i>
2 Kelompok	12,9307	28,8648	12,9307
3 Kelompok	41,7143	51,6196	41,7143
4 Kelompok	53,7953	53,7953	53,7953
5 Kelompok	42,2776	48,0143	50,5696
6 Kelompok	34,6190	46,9265	44,4247
7 Kelompok	31,1958	44,0721	41,8142
8 Kelompok	28,6931	44,3851	44,0585

Lampiran 27. Tabel Nilai Ratio Analisis Pengelompokan Data Kategorik

Nilai θ	Jumlah Kelompok	Nilai Ratio
0,07	2	0,1899
0,10	2	0,2027
0,12	2	0,1803
0,15	3	0,2504
0,17	3	0,1442
0,20	3	0,2272
0,22	2	0,1594
0,25	1	0,0000
0,27	2	0,1495

Lampiran 28. Tabel Nilai Ratio Analisis Ensemble

- a. Tabel Nilai ratio hasil pengelompokan ensemble ROCK

Nilai θ	Jumlah Kelompok	Nilai Ratio
0,07	4	0,1991
0,10	4	0,1991
0,12	4	0,1991
0,15	4	0,1991
0,17	4	0,1991
0,20	4	0,1991
0,22	4	0,1991
0,25	3	0,1732
0,27	3	0,1358
0,30	3	0,7000

- b. Tabel Nilai ratio hasil pengelompokan ensemble SWFM

Jumlah Kelompok	Nilai Ratio
2 Kelompok	0,3059
3 Kelompok	0,3639
4 Kelompok	0,3596
5 Kelompok	0,3727
6 Kelompok	0,3282

Lampiran 29. Output Software R untuk Pengujian Normalitas Multivariat

```
> mardiaTest(datanumerik, qqplot = TRUE)
Mardia's Multivariate Normality Test
-----
data : datanumerik

g1p          : 25.29325
chi.skew     : 105.3886
p.value.skew : 0.8266257

g2p          : 74.75091
z.kurtosis   : -1.037443
p.value.kurt : 0.2995296

chi.small.skew : 121.1414
p.value.small  : 0.4536444

Result       : Data are multivariate normal.
-----
```

Lampiran 30. Output Software R untuk Analisis Metode MANOVA

```
> manova (datanumerik~Data.Ensembl$theta.27)
Call:
  manova(datanumerik ~ Data.Ensembl$theta.27)

Terms:
              Data.Ensembl$theta.27 Residuals
resp 1                1.836    1060.698
resp 2                0.011      2.363
resp 3                1.492     16.348
resp 4                0.155      2.775
resp 5                4.905     225.615
resp 6                8.269     670.825
resp 7             6969.099  15363.779
resp 8                1.497      92.897
Deg. of Freedom              1          23

Residual standard errors: 6.790972 0.3205199 0.8430672 0.3473439
3.131985 5.400585 25.84551 2.009721
Estimated effects may be unbalanced

>
summary(manova (datanumerik~Data.Ensembl$theta.27), test="Wilks")
              Df    Wilks approx F num Df den Df    Pr(>F)
Data.Ensembl$theta.27  1 0.45167    2.428      8    16 0.06235
.
Residuals              23
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

“Halaman ini sengaja dikosongkan”

BIOGRAFI PENULIS

Nama : Alvionita
Tempat, Tanggal Lahir : Singkawang, 11 Februari 1993
Jenis Kelamin : Perempuan
Agama : Buddha
Alamat : Perumahan Taman Duta Mas
Blok C5 No. 1 RT/RW 04/02,
Kel. Baloi Permai, Kec. Batam
Kota, Kepulauan Riau,
Indonesia 29463
No. Telepon : 085668127484
Email : alvionita110293@gmail.com



Riwayat Pendidikan :

TKK BPK PENABUR Serang	(1997 - 1999)
SDK BPK PENABUR Serang	(1999 - 2001)
SDK BPK PENABUR Bogor	(2001 - 2003)
SDK Kalam Kudus Batam	(2004 - 2005)
SMPK Kalam Kudus Batam	(2005 - 2008)
SMA Katolik Yos Sudarso Batam	(2008 - 2011)
Statistika FMIPA Universitas Padjajaran (Sarjana)	(2011 - 2015)
Statistika FMIPA ITS Surabaya (Magister)	(2015 - 2017)